

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA
COMPUTAÇÃO**

Paula Renata dos Santos Ferreira

**Um Modelo Computacional da Teoria da Atividade
para Definição de Ambientes de Aprendizagem
Baseados em Framework Orientado a Objetos**

Dissertação submetida à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de Mestre em Ciência da Computação.

Prof. Leandro José Komosinski, Dr.

Florianópolis, Fevereiro de 2005

Um Modelo Computacional da Teoria da Atividade para Definição de Ambientes de Aprendizagem Baseados em Framework Orientado a Objetos

Paula Renata dos Santos Ferreira

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação, área de concentração Sistemas de Conhecimento e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

Raul Sidnei Wazlawick, Dr.

Banca Examinadora

Prof. Leandro José Komosinski, Dr.

Prof. Edla Maria Faust Ramos, Dra.

Prof. Ricardo Pereira e Silva, Dr.

Prof. Rafael Avila Faraco, Dr.

A arte de viver consiste em tirar o maior bem do maior mal.
Machado de Assis

Para minha mãe Malvina, que é o mais lindo presente de
Deus para mim.

Agradecimentos

A conclusão desta dissertação foi compartilhada por muitas pessoas especiais, a quais gostaria de agradecer intensamente.

Agradeço a Deus, meu senhor, que me deu coragem para seguir em frente e esteve sempre presente na minha vida. Agradeço pela própria vida e por todas as conquistas que Ele tem me proporcionado.

Agradeço aos meus pais Dario e Malvina pelo esforço e incentivo para que eu continuasse este projeto. Esses pais que eu amo e respeito e que são as pessoas mais importantes da minha vida. Obrigada por participarem tão efetivamente dela.

Agradeço aos meus irmãos Cristina, Anésia, Marcelo e Caique e sobrinhos Gustavo e Victoria por entenderem minha ausência, e sempre ficarem ao meu lado incentivando e apoiando meu mestrado. À minha família todo meu amor.

Agradeço ao meu namorado Paulo Henrique, por ter sido meu amor, meu amigo e meu companheiro em todos os momentos, tanto os de alegria quanto os de tristeza. Obrigada por todo carinho, dedicação, compreensão, pela força e por sempre estar ao meu lado.

Agradeço a Lisi, por ter sido e por ser, antes de tudo, amiga. Agradeço todos os incentivos, toda confiança e dedicação especial que sempre teve comigo. Pois dividiu comigo as angústias motivadas pela distância de casa e tantas outras coisas, como também as alegrias pelo nascimento desta nova amizade.

Agradeço ao Gubert(Guba), pois, foi mais que um amigo, ele sempre acreditou que eu conseguiria quando, que por muitas vezes, eu mesma não acreditava. Agradeço por todo incentivo, respeito e atenção que sempre teve comigo.

Agradeço ao Gilberto, pelo grande amigo que ele foi, como foi, e que ainda é. Amigo que eu aprendi a conhecer e amar. Esteve ao meu lado em momentos delicados e sempre tinha uma palavra de confiança e carinho.

Agradeço ao meu amigo Juvinski que esteve sempre disposto a me ajudar em tudo que eu precisasse, sendo por muitas vezes um ombro amigo e em outras, parte importante no desenvolvimento do meu trabalho.

Agradeço ainda aos meus amigos de café, discussões e aprendizado, Andréia e Alex, Ana e Francisco, Augusto e Giba, obrigada pela ótima convivência que me proporcionaram.

Agradeço ao meu orientador Leandro, por ter me ajudado a trilhar este caminho. Em cada reunião uma demonstração de sabedoria que por muitas vezes me enchia de esperança e me fazia ter forças para continuar. Agradeço também sua esposa Mônica, que abriu as portas da sua casa e sempre esteve disposta a ajudar no que eu precisasse.

Agradeço ao professor Dr. Rafael pela disposição de se deslocar até Florianópolis para prestigiar minha defesa, obrigada pelas sugestões que serviram para melhorar meu trabalho.

Agradeço ao professor Dr. Ricardo pela participação ativa no meu trabalho, sempre disposto a colaborar e, por ter sido o grande responsável por parte do conhecimento necessário para a realização deste.

Em especial agradeço a professora Dra. Edla que sempre me ajudou muito, me incentivando a lutar e continuar. Certamente é uma das grandes responsáveis pela conclusão do meu trabalho. A você todo meu agradecimento.

E finalmente e não menos importante, agradeço a Vera Lúcia, que sempre foi muito mais que a secretaria da pós-graduação, foi mãe e amiga para mim e certamente para muitos alunos que longe de casa e dos amigos tinham em nela tudo isso. Agradeço por tudo e por mais um pouco que você fez por mim. Te adoro.

Em fim, a todos, meu carinho, respeito e admiração.

Sumário

Lista de Figuras	x
Lista de Tabelas	xii
Resumo	xiii
Abstract	xiv
1 Introdução	1
1.1 O Uso do Computador na Educação	2
1.2 Bases Psicológicas da Aprendizagem	3
1.2.1 Tendência Comportamentalista	3
1.2.2 Tendência Construtivista	4
1.2.3 Tendência Sócio-Histórica	5
1.3 Problema	5
1.4 Objetivos	6
1.4.1 Objetivo Geral	6
1.4.2 Objetivos Específicos	6
1.5 Justificativa	7
1.6 Estrutura da Dissertação	7
2 Teoria da Atividade	9
2.1 Introdução	9
2.2 Breve Histórico	9

2.3	Níveis da Atividade	12
2.4	Estrutura da Atividade	15
2.4.1	Elementos Mediados	16
2.4.2	Elementos Mediadores	17
2.5	Princípios da Atividade	18
2.6	Teoria da Atividade na Educação	20
2.6.1	Ambientes educacionais baseados na Teoria da Atividade	21
2.7	Conclusão	26
3	Frameworks Orientado a Objetos	27
3.1	Introdução	27
3.2	Definição de Framework	28
3.3	Framework x Outros Conceitos	29
3.4	Benefícios dos Frameworks	31
3.5	Vantagens e Desvantagens	32
3.6	Frameworks Caixa-Branca e Caixa-Preta	33
3.7	Desenvolvendo Framework	34
3.8	Exemplos de Framework Orientado a Objetos	36
3.8.1	Framework FraG	36
3.8.2	Framework HyperToolBuilder	38
3.9	Conclusão	40
4	Frameworks para Ambientes Colaborativos	41
4.1	Introdução	41
4.2	O Framework Habanero	42
4.2.1	O Ambiente Habanero	43
4.3	O Framework Promondia	44
4.4	O Framework DARE	46
4.4.1	O Ambiente DARE	49
4.5	Conclusão	51

5	Um Modelo Computacional para a Teoria da Atividade	53
5.1	Introdução	53
5.2	Elementos da Teoria da Atividade	54
5.2.1	O Mediador Ferramenta	54
5.2.2	O Mediador Regra	55
5.2.3	O Mediador Divisão de Trabalho	57
5.3	Níveis da Teoria da Atividade	57
5.4	Associação da Estrutura e dos Níveis de Atividade	58
5.5	Conceitos de CSCL no MC	59
5.6	Implicações Educacionais no MC	60
5.7	Conclusão	61
6	Framework Proposto	62
6.1	Introdução	62
6.2	Descrição do FAAC	62
6.2.1	Diagrama Arquitetural do FAAC	63
6.3	Ambiente de Atividades Colaborativa	65
6.4	Conclusão	69
7	Conclusão	70
7.1	Trabalhos Futuros	71
	Referências Bibliográficas	72

Lista de Figuras

2.1	Relação Mediada	12
2.2	Níveis hierárquicos de uma atividade [Kuutti 1997]	13
2.3	Relacionamento mediado em nível individual [Engestrom e Miettinen 1999]	15
2.4	Estrutura básica de uma atividade[Engestrom e Miettinen 1999]	16
2.5	Dinosaur Canyon [Bellamy 1997]	22
2.6	Media Fusion[Bellamy 1997]	24
2.7	Nova Atividade [Komosinski 2000]	26
3.1	Aplicação Baseada em Framework [Mattsson 1996]	30
3.2	Relação custos x benefícios [Taligent 1994]	32
3.3	Processo do desenvolvimento do framework baseado em análise de domínio [Mattsson 1996]	35
3.4	Generalização do Domínio [Silva 2003]	36
3.5	Aplicação Jogo da Velha [Silva 2000]	37
3.6	Aplicação sob FraG [Silva 2000]	38
3.7	Diagrama da estrutura do framework HyperToolBuilder [Beutler 2003]	39
3.8	HyperBook [Beutler 2003]	40
4.1	Ferramentas do Habanero [Habanero 1996]	43
4.2	Ambiente Halbanero [Chabert 1997]	44
4.3	Chat [Gall e Hauck 1997]	45
4.4	<i>Whiteboard</i> Compartilhado [Gall e Hauck 1997]	46
4.5	Estrutura de uma atividade no DARE [Bourguin 2000]	47

4.6	Os conceitos da estrutura básica de uma atividade e o modelo conceitual do DARE [Bourguin 2000]	48
4.7	Exemplo de atividade [Bourguin 2000]	49
4.8	Exemplo da ferramenta [Bourguin 2000]	50
4.9	Exemplo da tarefa [Bourguin 2000]	51
5.1	Adaptação do Níveis da Atividade	54
5.2	Extensão da classe Ferramenta e Objeto	55
5.3	Extensão da classe Regra	56
5.4	Cadastro da Atividade	56
5.5	Triângulo da Atividade	58
5.6	Associação da Estrutura e dos Níveis da Atividade	59
6.1	Diagrama Arquitetural do FAAC	64
6.2	Cadastro da Atividade	66
6.3	Realizar Atividade	66
6.4	Resultado da Atividade1	67
6.5	Resultado da Atividade2	68
6.6	Histórico da Atividade1	68
6.7	Histórico da Atividade2	69

Lista de Tabelas

3.1	Bibliotecas de Classe X Framework [Taligent 1994]	29
-----	---	----

Resumo

As tecnologias educacionais sofreram evoluções desde o início do uso do computador no âmbito educacional. Atualmente tais ambientes são focados na aprendizagem colaborativa e ressaltam o uso do computador como mediador no processo de aprendizagem, o que vem ao encontro com a proposta da teoria utilizada nesta dissertação, a Teoria da Atividade.

Entretanto, para que estes softwares tenham um papel significativo dentro deste processo, é necessário o entendimento entre os profissionais envolvidos no desenvolvimento de um software educacional, que normalmente são de áreas distintas, a saber: computacional e pedagógica. Além desse entendimento, um outro fator abordado nesta dissertação foi o fato da Teoria da Atividade não possuir uma especificação para a área educacional, o que acaba dificultando sua projeção para técnicas computacionais.

Assim, foi proposto e desenvolvido um modelo computacional baseado na Teoria da Atividade. Tal modelo serve como elo entre a teoria e a técnica computacional utilizada. A partir dessa junção foi projetado um framework orientado a objetos para o desenvolvimento de ambiente de aprendizagem colaborativa.

Palavras-chave: Framework Orientado a Objetos, Teoria da Atividade, Informática na Educação.

Abstract

The educational technologies suffered evolutions since people started to use the computer with an educational purpose. These days, the mentioned surroundings are focused in the collaborative learning and they salient the computer use as a mediator in the learning process, which is related to the proposal theory utilized in this dissertation, the Activity Theory.

However, in order to these softwares be significant inside this process, it is necessary the understanding among the teachers involved in the educational software development, which is normally from distinct areas, such as: computational and pedagogical. Moreover, another aspect analyzed in this dissertation was the fact that the Activity Theory does not have a specification to the educational area, which makes difficult its projection to computational techniques.

Therefore, it was proposed and developed a computational model in the Activity Theory. This model is used as a link between the theory and the computational technical utilized. Since the junction, it was projected a framework oriented to objects to the development of the collaborative learning process ambient.

Keyword: Oriented Framework to Objects; Activity Theory; Educational Computing.

Capítulo 1

Introdução

A inserção do computador na rede escolar como ferramenta auxiliar no processo de ensino e aprendizagem tem potencial para provocar transformações no âmbito educacional. Estas mudanças ocorrem a partir do desenvolvimento de softwares educacionais utilizados em ambientes de aprendizagem.

Normalmente estes ambientes são espaços físicos no qual pessoas se encontram com a intenção de aprender. Esses ambientes estão sendo modificados pelo uso dos softwares educacionais, possibilitando a interação entre os estudantes e a utilização dos mesmos para construção do seu próprio conhecimento. O desenvolvimento destes softwares envolve duas áreas de conhecimento: a computacional e a pedagógica.

Entretanto, para que os softwares tenham um apelo educacional explícito, isto é, uma base pedagógica consistente, além de serem baseados em uma teoria de aprendizagem, é necessário que os profissionais destas áreas consigam unir seus conhecimentos em uma linguagem comum aos dois. Pois, os desenvolvedores de software tendem a enfatizar os aspectos operacionais do sistema.

Existem várias teorias psicopedagógicas de aprendizagem que são utilizadas na construção de softwares educacionais. No contexto desta dissertação, é adotada a Teoria da Atividade(TA). A TA é uma teoria psicológica que segue a corrente sócio-histórica, principalmente no que diz respeito a mediação por artefatos, que neste trabalho, será enfatizado pelo uso do computador como mediador do aprendizado.

A presente pesquisa busca aplicar alguns conceitos da Teoria da Atividade adaptada por Engeström para um Modelo Computacional (MC). Este modelo serve de ligação entre a teoria e a técnica computacional, utilizada para projetar um framework orientado a objetos para ambientes de aprendizagem. Um framework pode ser definido como “*um gerador de aplicações a partir de um determinado domínio*” [Mattsson 1996].

Assim, a partir do MC, os desenvolvedores de softwares educacionais para ambientes de aprendizagem podem ser auxiliados através da utilização do framework proposto.

1.1 O Uso do Computador na Educação

O projeto de implantação do computador nas escolas teve como pioneiros países como França e os Estados Unidos. Segundo [Valente 1999], nos EUA, o uso de computadores nas escolas foi pressionado pelo desenvolvimento tecnológico, pela necessidade de qualificação de profissionais, ou seja, pela necessidade de promover a alfabetização em informática, ou mesmo “automatizar o ensino”. Enquanto que na França, o objetivo era a preparação do aluno para ser capaz de usar a tecnologia da informática, não tendo como objetivo fundamental a mudança pedagógica.

No Brasil, o objetivo na utilização do computador era de promover algumas mudanças pedagógicas. Entretanto, o avanço pedagógico, tanto no Brasil, quanto nos EUA e na França, foram semelhantes e apontaram mudanças quase inexistentes do ponto de vista pedagógico [Valente 2000]. Ou seja, mesmo que a iniciativa brasileira tivesse esse cunho, o que se diferenciava dos EUA e da França, os resultados foram igualmente subsistentes.

Como a implantação da informática na educação nestes três países ocorreu aproximadamente no mesmo período, o Brasil teve uma grande influência dos EUA no que diz respeito aos primeiros softwares educacionais, inspirados no ensino programado com base na *Teoria Comportamentalista* e no condicionamento instrumental (estímulo-resposta). Portanto, os diversos softwares de instrução programada que foram implemen-

tados no computador no início dos anos 60 concretizavam a máquina de ensinar idealizada por Skinner no início dos anos 50 [Valente 1999] [Valente 2000].

1.2 Bases Psicológicas da Aprendizagem

Processos seculares como o ensino e a aprendizagem vêm sendo cada vez mais discutidos e repensados a partir do advento e da utilização das novas tecnologias da informação e da comunicação.

As tecnologias educacionais sofreram algumas evoluções desde do início do uso do computador no âmbito educacional. Nesta seção são apresentadas algumas teorias que podem ser utilizadas como referencial teórico no desenvolvimento de softwares educacionais. É abordado também como foi a uso do computador na educação e qual a tendência dos softwares educacionais vigentes.

1.2.1 Tendência Comportamentalista

A teoria comportamentalista deriva da corrente filosófica denominada empirismo, que enfatiza a experiência sensorial como fonte de conhecimento. Segundo [Davis e Oliveira 1994], seu grande defensor na psicologia foi o norte-americano Skinner. A teoria proposta por ele se preocupa em explicar os comportamentos observáveis do sujeito, desprezando a análise de outros aspectos da conduta humana como seu raciocínio, seus desejos e sentimentos.

As primeiras aplicações computacionais desenvolvidas, partiam do pressuposto de que a informação é a unidade fundamental no ensino, preocupando-se apenas em transmiti-la [Valente 1999]. Segundo [Aranha 1996], estes softwares, tradicionalmente conhecidos como CAI (Computer Aided Instruction), foram os primeiros desenvolvidos com intuito educacional e buscavam no comportamentalismo os procedimentos experimentais necessários para a aplicação do condicionamento e o controle do comportamento.

Para [Valente 1999], esta modalidade de software transpunha para o

computador o material didático funcionando basicamente como livros eletrônicos. Os alunos construíam seu conhecimento passo a passo através de estímulo e reforço, deixando pouco espaço para criatividade e ignorando os diferentes perfis destes alunos, que acabavam atuando de modo passivo.

1.2.2 Tendência Construtivista

Outra iniciativa de ordem pedagógica para o uso do computador, se deu através da criação da Linguagem Logo¹, baseada na *Teoria Construtivista*, que possibilitou a criação e o enriquecimento de ambientes educacionais [Valente 1999].

Na teoria construtivista, aprender significa construir um novo conhecimento, no qual o aluno passa de objeto para o sujeito da sua aprendizagem. Segundo [Davis e Oliveira 1994], Piaget, foi o teórico mais conhecido dessa visão, se dedicando a investigar cientificamente como se forma o conhecimento. De acordo com [Uchôa 2001], seu estudo é principalmente centrado em compreender como o aprendiz passa de um estado de "menor conhecimento" a outro de "maior conhecimento", o que está intimamente relacionado com o desenvolvimento pessoal do indivíduo.

Segundo Aranha, *"para os construtivistas, o conhecimento não é inato nem somente transmitido, não está só no sujeito e nem é dado apenas pelo objeto, mas se forma e se transforma pela interação entre ambos"* [Aranha 1996].

Os softwares educacionais apoiados na abordagem construtivista possibilitam aos estudantes terem uma maior flexibilidade na execução de suas atividades, aprendendo mais efetivamente e construindo seu próprio conhecimento, tendo um controle significativo da interação na aprendizagem [Valente 1999].

¹A linguagem de programação Logo foi criada em 1967 por Seymour Papert, tendo como base a teoria construtivista de Jean Piaget [Valente 1999].

1.2.3 Tendência Sócio-Histórica

Uma perspectiva emergente do uso da informática na educação se dá através do desenvolvimento de ambientes computacionais de aprendizagem cooperativa. Na interpretação de [Santos 1999], no desenvolvimento destes ambientes aplicam-se alguns princípios fundamentados na teoria psicológica *Sócio-Histórica*.

Segundo [Rego 1999], esta nova psicologia proposta por Vygotsky atribui uma enorme importância à dimensão social. Assim, o contexto social passa a ser considerado fundamental no processo de aprendizagem. Para [Santos 1999], embora a psicologia sócio-histórica tenha sido descoberta há pouco mais de 20 anos, pelos anglo-americanos, esta representa uma contribuição valiosa para o trabalho pedagógico nos dias atuais, redefinido as formas de aprendizagem colaborativas apoiadas ou não por computadores.

As aplicações computacionais voltadas para colaboração tendem a materializar os princípios definidos na teoria sócio-histórica. Segundo [Santos 1999], “*A discussão sobre redes de computadores, educação e cooperação passa necessariamente por Vygotsky*”.

Com o advento da internet, cada vez mais a ênfase educacional se volta para ambientes de aprendizagem colaborativa ou CSCL (*Computer Supported Collaborative Learning*). Esses ambientes ressaltam o uso do computador como mediador no processo de aprendizagem, o que vem de encontro com a proposta da Teoria da Atividade adaptada por Engeström que é apresentada no capítulo 2.

1.3 Problema

O desenvolvimento de software educacional de uma forma geral associa alguns problemas computacionais e pedagógicos. No âmbito computacional pode-se apontar o alto custo no desenvolvimento e na manutenção dos softwares, a preocupação com o prazo de entrega e a qualidade do produto final.

A questão pedagógica é apontada como outro problema relevante, pois se encontra implícita nos softwares educacionais. Pois, os desenvolvedores que não possuem conhecimento pedagógico tendem a enfatizar os aspectos operacionais dos softwares. O mesmo se dá com os profissionais da educação, que não possuem conhecimento na área computacional, dificultando o diálogo entre esses profissionais.

Cabe ressaltar que os processos de formação destes dois profissionais são distintos. Assim, para que o desenvolvimento de softwares educacionais tenha bases epistemológicas bem definidas é necessário o estabelecimento de um diálogo que denote o entendimento dos profissionais envolvidos, independente de suas áreas de formação. Entretanto, esse diálogo se constitui em um desafio a ser alcançado. Segundo [Candida 2002], *“os softwares educacionais disponíveis no mercado são baseados em perguntas e respostas e não dão oportunidade ao aluno de “construir o conhecimento” em geral recorrem ao aprendizado por repetição.”*

1.4 Objetivos

1.4.1 Objetivo Geral

Definir um Modelo Computacional baseado na Teoria da Atividade para construção de ambientes para aprendizagem colaborativa utilizando o conceito de frameworks orientados a objetos.

1.4.2 Objetivos Específicos

- Fazer uma adaptação da TA para servir ao modelo proposto.
- Projetar um Framework Orientado a Objetos para ambientes de aprendizagem colaborativa a partir do modelo computacional definido.
- Desenvolver um protótipo de um ambiente de aprendizagem colaborativa utilizando o framework proposto.

1.5 Justificativa

A utilização de softwares na educação tem aumentado consideravelmente a medida que a informática está mais presente na vida cotidiana. Normalmente estes softwares educacionais são desenvolvidos por profissionais da área computacional que acabam enfatizando mais os aspectos técnicos da aplicação por não ter formação acadêmica pedagógica ou porque tem pouco conhecimento sobre a aplicação de teorias psicopedagógicas.

No que diz respeito à questão pedagógica envolvida no desenvolvimento de softwares educacionais, uma alternativa para resolvê-la seria a definição de alguns conceitos da Teoria da Atividade em um Modelo Computacional.

A escolha da TA se dá pelo fato dos ambientes de aprendizagem vigentes caracterizarem o uso do computador como mediador da aprendizagem, bem como a questão da colaboração.

A fim de facilitar o entendimento dos desenvolvedores de softwares educacionais optou-se por utilizar a técnica de framework orientado a objetos, tendo como domínio o Modelo Computacional baseado na TA.

A utilização do conceito de frameworks, nesta dissertação, visa a reutilização pelas aplicações que podem ser desenvolvidas sob ele. Além de uma provável diminuição de falhas e erros, a característica do seu domínio é fundamentado em uma teoria de aprendizagem, o que determina sua intenção educacional.

1.6 Estrutura da Dissertação

No capítulo 2 são descritos os princípios básicos da Teoria da Atividade, a estrutura da atividade humana proposta por Leont'ev, bem como a adaptada por Engeström. Também neste capítulo são apresentados três softwares aplicados na educação que utilizam alguns conceitos da TA.

O capítulo 3 aborda alguns conceitos de Framework Orientado a Objeto.

tos, suas características, vantagens e desvantagens de sua utilização e dois exemplos de frameworks orientados a objetos para um melhor entendimento do conceito.

O capítulo 4 apresenta alguns frameworks baseados em ambientes colaborativos e descreve algumas possibilidades de uso destes na aprendizagem.

O capítulo 5 apresenta um Modelo Computacional (MC) que foi desenvolvido a partir da análise de alguns conceitos da Teoria da Atividade (TA).

O capítulo 6 descreve o projeto do framework proposto que foi definido pelas considerações do Modelo Computacional, bem como, um protótipo desenvolvido a partir deste framework.

Capítulo 2

Teoria da Atividade

2.1 Introdução

O objetivo deste capítulo é apresentar um breve histórico do surgimento da Teoria da Atividade, a estrutura proposta por Leont'ev, bem como a adaptação de Engeström. Será abordado ainda, os princípios que definem esta teoria, os níveis da atividade humana e as implicações educacionais que podem ser utilizadas a partir desta.

Embora a TA seja uma teoria psicológica, esta pode ser utilizada como base pedagógica no desenvolvimento de softwares educacionais. A estrutura da atividade de Engeström representa uma forma de relacionamento colaborativo, o que vem de encontro com a proposta dos software vigentes que tendem a enfatizar a colaboração no processo de ensino e aprendizagem.

Como a proposta desta pesquisa é modelar esta teoria para uma visão computacional. Este capítulo se torna relevante para uma maior compreensão da mesma.

2.2 Breve Histórico

A teoria sócio-histórica criada em 1920 por Lev Semenovitch Vygotsky e seus colaboradores, Alexis Leont'ev e Alexander Romanovich Luria, tinha como objetivo criar uma nova psicologia, algo que confrontasse com às tendências que predomina-

vam na época.

Desse modo, existia de um lado um grupo baseado em pressupostos da filosofia empirista, que segundo [Rego 1999], “*via a psicologia como ciência natural que devia deter na descrição das formas exteriores de comportamento, entendidas como habilidades mecanicamente construídas, procurando explicar apenas os processos elementares sensoriais e reflexos do ser humano*”.

O outro grupo, inspirado nos princípios da filosofia idealista, entendia a psicologia como ciência mental que descrevia as propriedades dos processos psicológicos superiores. Entretanto, acreditava que a vida psíquica humana não poderia ser objeto de estudo da ciência objetiva, já era manifestação do espírito [Oliveira 1993].

Para Vygotsky nenhuma das tendências explicava de maneira satisfatória o desenvolvimento das funções psicológicas da espécie humana, pois, para ele, o desenvolvimento biológico é formado e concretizado por meio do desenvolvimento social e histórico. Assim, ele definiu uma psicologia que abordasse o homem “*enquanto corpo e mente, enquanto ser biológico e ser social, enquanto membro da espécie humana e participante de um processo histórico*” [Oliveira 1993].

Em busca desta nova psicologia, Vygotsky dedicou-se principalmente ao estudo das funções psicológicas superiores, se interessando em compreender os mecanismos mais sofisticados do ser humano, como a capacidade de planejar ações que serão realizadas posteriormente e a intencionalidade destas. Segundo [Oliveira 1993], esta capacidade de abstração e planejamento “*se difere dos processos elementares como as ações puramente reflexas e automatizadas, como por exemplo, a sucção da mamadeira pelo bebê, ou mesmo o ato de movimentar a cabeça na direção de um barulho repentino*”.

Para estudar estes processos, Vygotsky e seus colaboradores estudaram o comportamento dos animais mais capazes e identificaram que a intencionalidade é uma característica tipicamente humana. [Oliveira 1993] descreve um exemplo no qual fica claro a diferença dos processos elementares e do que Vygotsky chama de processos psicológicos superiores, ou mesmo a intencionalidade da ação.

...“é possível ensinar um animal a acender a luz num quarto escuro. Mas o animal não seria capaz de, **voluntariamente**, deixar de realizar o gesto aprendido porque vê uma pessoa dormindo no quarto”.

Vale a pena ressaltar que, o que diferencia a atividade do animal e do homem é a intencionalidade na realização da atividade, no exemplo apresentado, o ato de acender a luz. A atividade animal é caracterizada pela relação direta entre o motivo e o objeto da atividade. Deixar de acender a luz a partir de uma nova informação (que é alguém dormindo no quarto) é uma percepção de caráter voluntário tipicamente humano. Essa decisão deixa de ser direta e passa a ser *mediada* pela ação consciente do sujeito com seu objeto.

Em [Moll 1996] é descrito que, “o postulado fundamental abordado por Vygotsky é que as funções psicológicas humanas diferem dos processos psicológicos de outros animais porque são culturalmente mediadas, historicamente desenvolvidas e emergem da atividade prática”.

Nessa perspectiva, Leont’ev passou a considerar as atividades humanas como formas de relação do homem com o mundo, sendo estas dirigidas por motivos e por fins a serem alcançados [Oliveira 1993]. Pode-se dizer que, só existe uma atividade se esta for movida por um objetivo e por meios de ações planejadas. Entretanto é plausível dizer que, estas ações se referem à intencionalidade da atividade que diferencia os processos psicológicos superiores proposto por Vygotsky. Assim emergiu a Teoria da Atividade. A TA é dividida em três gerações, a saber:

- A primeira iniciada com os trabalhos de Vygotsky que dizem respeito a mediação por artefatos;
- A segunda na qual Leont’ev dá continuação às descobertas de Vygotsky;
- A terceira a adaptação realizada por Engeström na estrutura da atividade.

Sobre a primeira geração pode-se dizer que, nos primeiros trabalhos realizados pela escola sócio-histórica de Vygotsky, Leont’ev e Luria, “a unidade de análise

era a ação orientada por objetivos, sendo a mediação feita através de ferramentas culturais e sinais” [Engestrom e Miettinen 1999].

Toda forma elementar de comportamento pressupõe uma reação direta à situação-problema defrontada pelo organismo - o que pode ser representado pela fórmula simples (S - R). Por outro lado, a estrutura de operações com signos requer um elo intermediário entre o estímulo e a resposta. Esse elo intermediário é um estímulo de segunda ordem (signo), colocado no interior da operação, onde preenche uma função especial; ele cria uma nova relação entre S e R. O termo "colocado" indica que o indivíduo deve estar ativamente engajado no estabelecimento desse elo de ligação. Esse signo possui, também, a característica importante de ação reversa. Conseqüentemente, o processo simples estímulo-resposta é substituído por um ato complexo, mediado, que representamos da seguinte forma [Vygotsky 1984] 2.1:

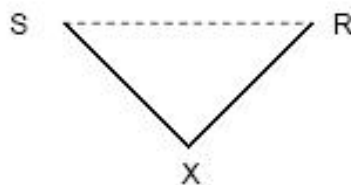


Figura 2.1: Relação Mediada

Na segunda geração Leont'ev argumenta que apenas a ação individual pelo sujeito não é suficiente, sugerindo a interação de outras pessoas na atividade. Sua maior contribuição pode ser destacada através dos níveis de atividade proposto por ele como: a própria atividade, a ação e a operação.

2.3 Níveis da Atividade

O termo atividade é conceituado por [Leontiev 1978] como “os processos que são psicologicamente determinados pelo fato de aquilo para que tendem no seu

conjunto(o seu objeto) coincidir sempre com o elemento objetivo que incita o paciente a uma dada atividade, isto é, o motivo”. Sendo assim, uma atividade é sempre motivada pelo objetivo de alcançar algo, aprender algo.

Na figura 2.2 será apresentada a relação entre os níveis da atividade humana, na qual, atividade, ação e operação significam, motivo, objetivo e condições instrumentais, respectivamente.

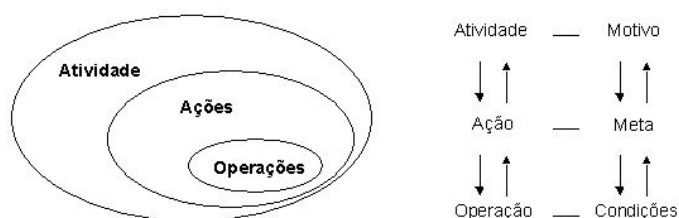


Figura 2.2: Níveis hierárquicos de uma atividade [Kuutti 1997]

Um exemplo apresentado em [Leontiev 1978] se refere a um aluno que esta lendo um livro para realizar um exame. Mas a que está dirigida sua atividade? Se ao saber que este livro que ele está lendo não será mais necessário para a realização do exame, pode ocorrer uma falta de vontade imediata na continuação da leitura do livro. O fato do aluno estar lendo o livro, não significa que a sua atividade é de obter conhecimento sobre o conteúdo do livro mas sim de ter êxito no exame. Sendo assim, o motivo da sua atividade não coincide com o resultado almejado, pois ...“*nem toda **atividade** é uma ação...*”. Entretanto, uma atividade pode ser realizada por ações individuais ou coletivas.

Uma **ação** é um processo cujo motivo não coincide com o seu objeto(com o que ele visa, seu objetivo), pois pertence à atividade em que entra a ação considerada. No exemplo do estudante que parou a leitura ao saber que não seria mais necessária para o exame, trata-se de uma ação, ou seja, ...“*a ação é o fim imediato concretizado*’ [Leontiev 1978]’.

No exemplo do estudante o fim da leitura significaria apropriar-se do conteúdo e este fim imediato mantém uma relação determinada com o motivo da atividade

que é passar no exame. Podemos dizer que, a ação é planejamento consciente de uma determinada atividade e que sua realização se dá através de operações. A operação é a maneira como será executada a ação, é o conteúdo indispensável de toda ação.

Para [Leontiev 1978], “*Enquanto uma ação é determinada pelo seu fim, as operações dependem das condições em que é dado este fim*”. Para [Kuutti 1997], uma **operação** é a forma prática de executar uma ação, isso ocorre quando o indivíduo não precisa mais pensar em como realizar a atividade, pois as ações já estão internalizadas nele.

Outro exemplo clássico de Leont’ev citado por [Engestrom e Miettinen 1999] para esclarecer o conceito de atividade e dos três níveis que a compõe é o exemplo de uma caçada coletiva primitiva.

Quando um membro de um grupo realiza sua atividade de trabalho, ele também o faz para satisfazer uma de suas necessidades. Um batedor, por exemplo, tomando parte de uma caçada coletiva primitiva, foi estimulado por uma necessidade de vestuário (...) O que, entretanto, essa atividade visava diretamente? Ela poderia ter visado, por exemplo, a assustar um rebanho de animais e mandá-los na direção de outros caçadores, escondidos para emboscá-los. Isso (...) é o que deveria ser o resultado da atividade desse homem. E a atividade desse membro individual da caçada termina com isso. O resto é completado pelos outros membros. Esse resultado, compreensivelmente não leva e não pode levar, em si mesmo, à satisfação da necessidade do batedor por comida ou pela pele do animal. Os alvos dos processos de sua atividade, conseqüentemente, coincidiram com o que os estimulou, não coincidiram com os motivos de sua atividade.

As ações realizadas pelos membros da caçada embora distintas e as vezes até parecendo sem sentido quando olhada isoladamente tornam-se significativa quando analisada como um todo.

Na seção 5.2 será apresentada a estrutura da atividade proposta por Le-

ont’ev e a estrutura da atividade proposta por Engeström.

2.4 Estrutura da Atividade

A estrutura da atividade no nível individual, representa a mediação através da relação entre o sujeito e objeto. Esta relação é sempre mediada por vários artefatos, sejam eles concretos (instrumentos, máquinas, etc.) ou abstratos (sinais, procedimentos, leis, etc). Estes artefatos irão se transformando durante o desenvolvimento da própria atividade e trazendo consigo uma cultura particular, restos históricos daquele desenvolvimento [Kuutti 1997]. A Figura 2.3 representa uma estrutura em um nível individual, na qual a ferramenta faz o papel de mediador entre o sujeito e o objeto.

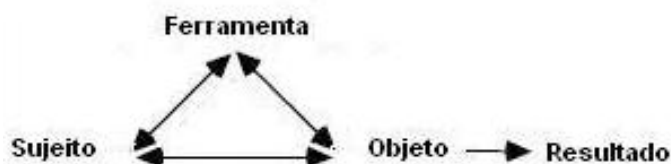


Figura 2.3: Relacionamento mediado em nível individual [Engestrom e Miettinen 1999]

[Duarte 2004] ressalta que, “*um objeto cultural, seja ele um objeto material, como por exemplo um utensílio doméstico, seja ele um objeto não material como uma palavra, tem uma função social, tem um significado socialmente estabelecido e deve ser empregado de uma determinada maneira*” .

Mediações por outros sujeitos e relações sociais não foram teoricamente integradas no modelo triangular da ação, apresentado na Figura 2.3. Entretanto, Leont’ev descobriu a integração entre a atividade coletiva e a individual emergindo assim a relação da divisão do trabalho. Como descrito em [Engestrom e Miettinen 1999], o desenvolvimento central das idéias da TA pressupõe um estudo cuidadoso das obras de Marx.

A partir do trabalhos de Leont’ev, Engeström redefiniu a estrutura da

atividade adicionando a comunidade e suas relações, regras e divisão de trabalho. Esta pode ser considerada a terceira geração da TA, bem como, seu estado da arte. Na Figura 2.4 apresenta-se como essa relação ocorre.

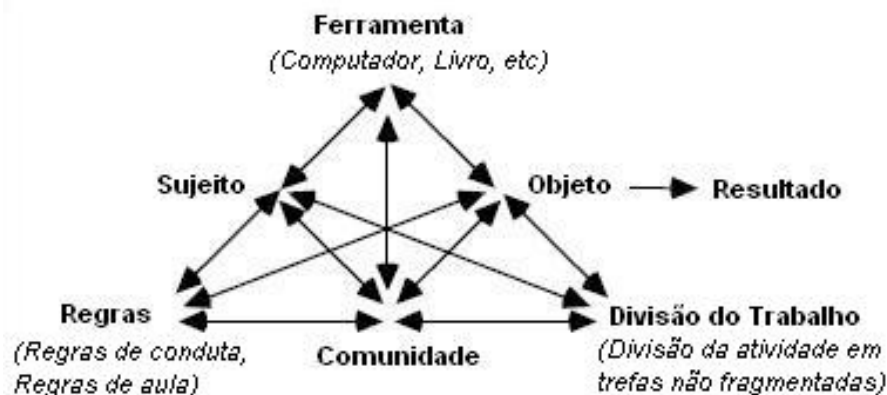


Figura 2.4: Estrutura básica de uma atividade[Engeström e Miettinen 1999]

O modelo apresentado pode ser representado por dois grupos, os elementos mediadores e os elementos mediados.

2.4.1 Elementos Mediados

Os elementos mediados são representados pelo Sujeito, pela Comunidade e pelo Objeto.

2.4.1.1 Sujeito

O *sujeito* atua sobre o objeto através das ferramentas (artefatos mediadores) que estão disponíveis na atividade. Ele pode executar uma parte da atividade individualmente ou se inserir em uma comunidade e participar da atividade coletiva proposta por esta.

2.4.1.2 Comunidade

A **Comunidade** na Teoria da Atividade tem um papel fundamental, pois desde o nosso nascimento já estamos inseridos em uma comunidade. Segundo [Rovai 2002], comunidade é “..um grupo de pessoas que são socialmente interdependentes, que participam em conjunto de discussões e na tomada de decisões, e que compartilham certas práticas que ao mesmo tempo definem a comunidade e são nutridas por ela...”.

2.4.1.3 Objeto

Toda atividade deve ter um **Objeto** para ser transformado. Este objeto pode ser físico ou abstrato. O que determina o objeto da atividade é o contexto que este objeto está inserido.

2.4.2 Elementos Mediadores

Os elementos mediadores são representados pela Ferramenta, pelas Regras e pela Divisão de Trabalho.

2.4.2.1 Ferramenta

O papel da **Ferramenta** em uma atividade é descrição conceituada por Vygotsky sendo a responsável pela mediação entre o sujeito e o objeto. Assim, as propriedades destes artefatos é fundamental para a transformação do objeto.

2.4.2.2 Regras

As **regras** nos termos apontados por [Moll 1996], são consideradas estruturas internas de práticas que foram desenvolvidas historicamente para responder as necessidades do ambiente físico ou social e evoluem de acordo com sua própria dinâmica. Podendo assim, organizar a vida social da comunidade.

2.4.2.3 Divisão de Trabalho

A *Divisão de Trabalho* é realizada pela comunidade. As tarefas são distribuídas de uma maneira não fragmentada, na qual todos os membros possam participar da atividade toda, envolvendo tanto trabalho físico quanto o intelectual.

2.5 Princípios da Atividade

A TA é considerada como uma teoria multidisciplinar que pode ser utilizada em qualquer campo de atuação humana. Os princípios que fundamentam esta teoria podem ser encontrados em [Kaptelinin e Nardi 1997][Bannon 1997][Engestrom e Miettinen 1999]:

- *Princípios da Orientação a Objetos*¹.

Segundo [Bannon 1997] este princípio “*é um dos mais importantes da teoria da atividade. Entretanto, é também o mais controverso e potencial de confundir*”. Este descreve o ponto de vista específico da TA na natureza dos objetos com a qual os seres humanos interagem. Entretanto, o princípio de orientação a objetos indica que os seres humanos vivem em uma realidade que seja objetiva em um sentido amplo, as coisas que constituem esta realidade têm não somente as propriedades que são consideradas objetivas de acordo com ciências naturais mas propriedades definidas social e culturalmente.

- *Princípios da Estrutura Hierárquica da Atividade*

Este princípio divide o processo da atividade em três níveis distintos, que são, *atividade, ação e operação*, descritos na seção 2.3 deste capítulo.

- *Princípios da internalização e externalização*

¹A idéia do “objeto” não é limitada às propriedades físicas, químicas e biológicas das entidades, este “objeto” abrange *também* propriedades social e culturalmente determinadas. Não tem significado nenhum como programação orientada a objetos [Bellamy 1997]

As atividades internas não podem ser compreendidas se forem analisadas separadamente das atividades externas, porque há uma transformação mútua entre estes dois tipos de atividades.

A internalização é o processo de absorção das atividades externas nas internas, fornece uma possibilidade para seres humanos para simular interações potenciais com realidade sem executar manipulações reais em objetos reais, ou seja, está relacionada com a reprodução da cultura: o ser humano internaliza conhecimentos, conceitos, valores e significados, reproduzindo-os em suas relações sociais.

A externalização de forma contrária da internalização, transforma as atividades internas e externas. Segundo [Engestrom e Miettinen 1999], a externalização está ligada à capacidade criativa do ser humano, através da qual é possível transformar a realidade vivida. No processo de externalização, poderão ser criadas novas ferramentas técnicas ou psicológicas com o papel de mediadoras na relação entre sujeito e objeto, potencializando a superação do processo de reprodução cultural e caracterizando um ciclo expansivo de desenvolvimento. Assim, as representações internas se transformam em externa através da fala, do gesto, da escrita, da manipulação disponível no meio ambiente e da mesma maneira os processos internos se externalizam.

- *Princípios da Mediação*

A TA enfatiza que a atividade humana é mediada através de ferramentas que, segundo Vygotsky, podem ser de dois tipos: as técnicas e as psicológicas. As técnicas manipulam objetos físicos (por exemplo, um computador) enquanto os psicológicos podem ser usados para influenciar a si mesmo (por exemplo, como signos, calendários). Estas ferramentas, são criadas e transformadas durante o próprio desenvolvimento da atividade e carregam com elas uma cultura particular. Assim, o uso das ferramentas se transforma em meios para a acumulação e a transmissão do conhecimento social, influenciando a natureza, não somente do comportamento externo, mas também de funcionar mental dos indivíduos.

Segundo [Oliveira 1993], “*a mediação é um processo essencial para tornar possível atividades psicológicas voluntárias, intencionais, controladas pelo próprio indivíduo*”.

- *Princípios do Desenvolvimento*

A teoria da atividade requer que a interação do ser humano com a realidade, seja analisada dentro do contexto do desenvolvimento, pois, com o passar do tempo ele sofre alterações e entender essas alterações ajudará no entendimento do seu estado atual. De acordo com [Leontiev 1978], “*...cada indivíduo aprende a ser um homem. O que a natureza lhe dá quando nasce não lhe basta para viver em sociedade. É-lhe ainda preciso adquirir o que foi alcançado no decurso do desenvolvimento histórico da sociedade humana*”.

O homem, por meio de sua atividade sobre o mundo material, humaniza o mundo, isto é, ao mesmo tempo em que atua e trabalha, suas aptidões e conhecimentos vão se cristalizando, de certa maneira, nos seus produtos. Pensemos como a atividade de escrever, inventada pelo homem em um processo de milhares de anos, criou a habilidade de escrever e inventou, ao mesmo tempo, o lápis, a caneta, ou o pincel. Pensemos, agora, como esses objetos carregam em si a habilidade criada [BOCK 2004].

2.6 Teoria da Atividade na Educação

A formação do indivíduo é sempre um processo educativo, podendo este ser direto ou indireto, intencional ou não intencional, realizado por meios de atividade práticas ou de explanações orais, etc. **Newton Duarte**

Segundo [Bellamy 1997], “*a teoria de desenvolvimento humano de Vygotsky tem um número de implicações para educação. Entre elas se pode citar: porque o pensamento é mediado por artefatos, e os estilos de pensar que são exibidos e são baseados na cultura, e esta cultura é baseada nas ferramentas usadas dentro dela*”. Isto sugere

que, se a ajuda na educação é ativamente engajada na cultura, aprendizes podem usar artefatos utilizados por especialistas da área, bem como participar de atividades similares as de especialistas.

Se o entendimento é mediado socialmente, os aprendizes devem estar engajados em discussões e debates com a comunidade que consiste em especialistas e outros aprendizes. Nesta situação, professores, especialistas e estudantes podem aprender um com o outro, fazendo com que os estudantes evoluam. Por isto a ênfase no aprendizado colaborativo é cada vez maior.

Estudantes engajados em uma comunidade de diferentes níveis de habilidades trabalham juntos para compartilhar o mesmo objetivo. Os que possuem menos habilidades trabalham lado a lado com os especialistas, e esta interação faz com que os estudantes de menor habilidade com o tempo se tornem experientes. Sendo assim, com a ajuda de pessoas mais experientes, os indivíduos podem desenvolver habilidades que antes estes eram incapazes de realizar. Essa performance é chamada por Vygotsky de zona de desenvolvimento proximal, que é a distância entre o que o indivíduo consegue realizar sozinho e o que ele pode aprender com a ajuda de pessoas mais experientes.

2.6.1 Ambientes educacionais baseados na Teoria da Atividade

Os ambientes apresentados nesta seção, tem características que foram baseadas na da Teoria da Atividade e nos conceitos básicos da teoria sócio-histórica, como a colaboração, o engajamento no projeto e a construção do conhecimento.

2.6.1.1 Dinosaur Canyon

O Dinosaur Canyon, apresentado por [Bellamy 1997] foi projetado para ensinar ciências da terra. Este software é a simulação de um canyon, um laboratório de petrologia² e paleontologia³, e foi projetado para os estudantes que não tem acesso direto a um canyon verdadeiro e nem como coletar fósseis e pedras reais, além de não terem um

²Parte da Geologia que se ocupa das rochas.

³Ciência que trata dos fósseis (animais e vegetais).

laboratório para analisarem suas amostras. No Dinosaur Canyon os alunos trabalham em pequenos grupos, cada equipe estuda uma porção de uma sequência geológica através do canyon. Dentro dessas porções eles coletam amostras de fósseis e rochas que são analisadas dentro de seus laboratórios específicos. No laboratório de petrologia as pedras são estudadas com o auxílio de um microscópio petrológico, obtendo dados precisos sobre sua formação e os elementos que a constituem. No laboratório de paleontologia os estudantes podem ver e medir os fósseis. Exemplo na Figura 2.5.

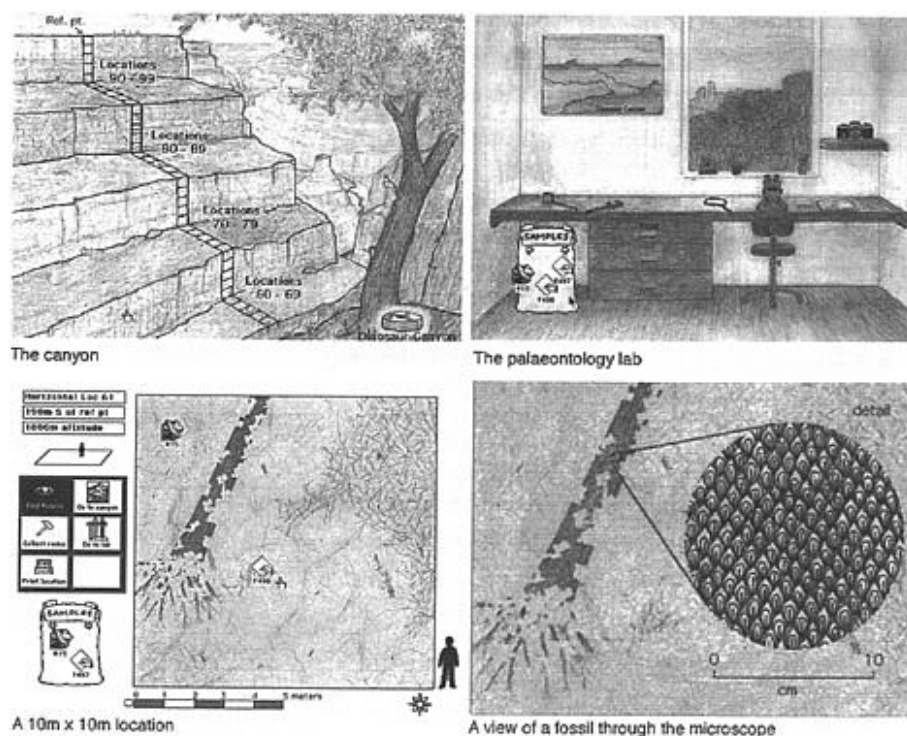


Figura 2.5: Dinosaur Canyon [Bellamy 1997]

O Dinossaur Canyon suporta três princípios que compõe a teoria sócio-histórica de Vygotsky: colaboração, construção e atividades autênticas.

O suporte para colaboração é um princípio importante para um ambiente educacional derivado da TA. Essa é aplicada no software quando um grupo estuda um conjunto de lugares, buscando a interpretação coerente no canyon. Uma outra forma de colaboração disponível, são os geólogos especialista que estão disponíveis para troca de

informações com os estudantes.

A construção é usada de forma que os estudantes construam a sua interpretação do canyon usando descobertas do programa e da literatura. Cada estudante apresenta suas descobertas da mesma categoria de estudo, e o professor escolhe uma pesquisa sobre qualquer aspecto do canyon para que os estudantes escolham para estudar em detalhes. No que diz respeito a atividade autêntica, os estudantes se engajam nas mesmas atividades que os especialistas da área.

Fazendo uma analogia entre os princípios definidos no Dinosaur Canyon com a estrutura da atividade de Engeström apresentada neste capítulo pode-se considerar que a colaboração e a atividade autêntica esta localizada no conceito de comunidade, na qual estão representados os sujeitos, os professores e os especialistas envolvidos em cada atividade. O estudo de uma atividade em grupo fica representado na divisão de trabalho, na qual os membros da comunidade se dividem para executar uma parte do trabalho para depois juntar as considerações para compreensão da atividade toda. A construção pode ser identificada nos níveis da atividade apresentados neste capítulo, na qual uma ação é o planejamento de cada aluno e a operação a maneira que cada um irá interpretar a atividade.

O Dinosaur Canyon foi projetado especificamente para permitir aos estudantes se envolverem nas mesmas atividades que especialistas da área. Durante as aulas em que o software era usado, os estudantes mostraram muito mais interesse em gerenciar seu trabalho e buscar interpretações mais apropriadas para suas pesquisas.

2.6.1.2 Media Fusion

Media Fusion, descrito em [Bellamy 1997], é uma ferramenta que busca apoiar a aprendizagem, unindo vídeo e ferramentas de análise de dados. O Media Fusion permite aos estudantes explorarem assuntos como o efeito estufa. Exemplo na Figura 2.6.

Da mesma maneira que o Dinosaur Canyon, o Media Fusion também busca utilizar alguns conceitos da teoria sócio-histórica de Vygotsky. Usando esta ferramenta os estudantes podem criar suas próprias mensagens de vídeo especificando várias

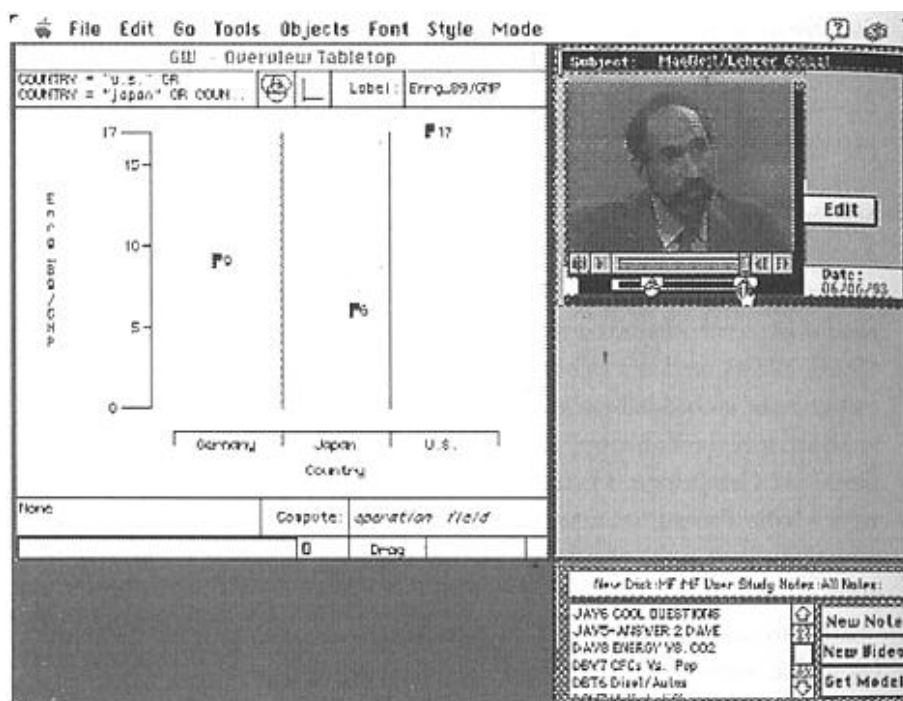


Figura 2.6: Media Fusion[Bellamy 1997]

análises dos dados. Eles trabalham em pequenos grupos e podem se comunicar com outros estudantes ao redor do país, enfatizando a colaboração do grupo envolvido na atividade.

Para a construção do conhecimento os estudantes constroem uma interpretação do domínio através da análise de dados, compartilhando suas análises e discutindo sobre elas com outros alunos.

Na atividade autêntica, os estudantes investigam assuntos que são importantes atualmente para o mundo. Usam dados do mundo real nos eventos correntes, e tem acesso a argumentos apresentados para o governo e as discussões correntes entre os cientistas.

2.6.1.3 Sam Cat

O Sam Cat, desenvolvido por ⁴ [Komosinski 2000] foi projetado para mostrar como os conceitos psicológicos e educacionais, especialmente ligados a Artefatos Mediadores de Aprendizagem podem ser incorporados em um software educacional. Estes conceitos estão diretamente relacionados com a Teoria da Atividade.

A finalidade do Sam Cat é apoiar o estudante na construção da sua Identidade Epistemológica (IE), que de acordo com [Komosinski 2000], “*IE é definida como uma postura intencional diante do conhecimento.*” Esta “postura intencional” no contexto da TA se refere a ação que o indivíduo tem antes de transformar seu objetivo em resultado, sendo o planejamento consciente que leva a realização das operações.

Cada aluno cadastrado no Sam Cat possui uma página pessoal na qual serão armazenados todos os registros das atividades desenvolvidas por ele. O estudante escolhe qual atividade irá realizar a partir de sua própria percepção de necessidade ou sob mediação do professor.

A atividade “nova atividade” que será apresentada na Figura 2.7 o estudante é levado à refletir sobre suas estratégias de aprendizagem. De acordo com Komosinski, “*tais estratégias, no contexto habitual, passam despercebidas e são usadas mecanicamente.*”. Com a atividade proposta no Sam Cat o estudante “despertará sua consciência” para o contexto no qual está inserido. Assim, aprender passa ser um ato intencional ajudando o aluno na construção da sua IE. Os elementos da atividade apresentados induzem os estudantes a pensar em cada um dos elementos envolvidos na sua atividade de aprendizagem específica como se fossem elementos independentes. Cabe então ao mediador, geralmente o professor, a tarefa de manter a unidade.

⁴Software de Apoio à Mediação do Centro para Aprendizagem Tecnológica

Leandro's Web SAM Site

Nova Atividade

Definir uma nova atividade implica em identificar os elementos que a compõem. Para isso, preencha o formulário abaixo. Se necessário, clique [aqui](#) para relembrar a definição de atividade.

☐ Mediar ☐ Não Mediar

Nome da Atividade:

Objeto:

Resultado:

Comunidade:

Ferramentas:

Figura 2.7: Nova Atividade [Komosinski 2000]

2.7 Conclusão

A Teoria da Atividade foi escolhida para fundamentar este trabalho pois apresenta o conceito de mediação por artefatos, a colaboração, e a construção do conhecimento, além de possibilitar implicações para a educação. Esta teoria vem de encontro com a proposta desta dissertação, pois, os conceitos citados acima são inerentes ao desenvolvimento de ambientes de aprendizagem colaborativa.

Assim, este capítulo apresentou a origem e o histórico da Teoria da Atividade, expondo os níveis, a estrutura e os princípios da mesma. A seguir foi descrito sobre a implicação da TA no âmbito educacional, e por fim, três exemplos de softwares educacionais baseados nesta teoria.

Capítulo 3

Frameworks Orientado a Objetos

3.1 Introdução

Com a evolução tecnológica, a informática começou penetrar na sociedade tornando-se parte do seu cotidiano. Isso impulsionou a criação e o desenvolvimento de novos softwares que, ao longo do tempo, vem se tornando mais complexos devido a novas descobertas da área de hardware e evolução das linguagens de programação.

Com o aumento da demanda por novos sistemas computacionais houve a necessidade de criar ou aperfeiçoar novas técnicas de desenvolvimento afim de conter a crise do software. Segundo [Pressman 1995], a palavra “crise” esta associada com um conjunto de problemas referentes ao desenvolvimento de software computacionais. Alguns problema citados por [Pressman 1995] são: o fracasso nas estimativas de prazo e o custo do software, a insatisfação do produto final, a dificuldade de manutenção, dentre outros. A solução se deu através do uso de técnicas, métodos e ferramentas que buscavam maior qualidade, produtividade e principalmente o reuso, desde o projeto até o desenvolvimento de softwares.

No fim dos anos 80, uma nova técnica chamava a atenção dos pesquisadores; o conceito de framework orientado a objetos [Bosch et al. 1997]. Este fornece uma solução para uma família de problemas, por meio de um modelo abstrato que pode ser personalizado e ajustado às necessidades específicas das aplicações desenvolvidas so-

bre ele. Segundo a empresa [Taligent 1995], os frameworks fornecem uma infraestrutura e uma interface flexível aos desenvolvedores de software. Na seção 3.2 serão apresentadas algumas definições e conceitos que envolvem o desenvolvimento e uso de frameworks.

3.2 Definição de Framework

Existem várias definições para frameworks. Entretanto uma delas consistem em dizer que *“um framework é um projeto reusavel de todo ou parte de um sistema de software descrito por um conjunto de classes abstratas e os meios pelas quais essas colaboram”* [Roberts e Johnson 1996] [Johnson 1997].

Uma outra definição é de que *“um framework é o esqueleto de uma aplicação que pode ser adaptada por um desenvolvedor de aplicação”*. Segundo [Johnson 1997], a primeira definição não é conflitante com a segunda, pois uma descreve a estrutura de um framework, enquanto a outra o seu propósito, sendo assim, um framework consiste em uma estrutura semi-acabada sobre a qual um desenvolvedor poderá construir uma nova aplicação sem iniciar o desenvolvimento do zero.

Usando as palavras de [Mattsson 1996] para uma maior compreensão do conceito, *“um framework pode ser visto como gerador de aplicações desde que seja planejado para ser usado como base para o desenvolvimento de várias aplicações dentro de um determinado domínio”*.

Porém, de acordo com a empresa [Taligent 1995], para que possa ser considerado um framework, este deve possuir duas características inerentes a seu conceito, a saber:

- **Os framework fornecem infraestrutura e projeto:** provendo uma infraestrutura(código), o framework diminui significativamente a quantidade de códigos que o desenvolvedor terá que programar, testar e depurar.
- **Inversão de Controle**¹: o papel do framework é fornecer o fluxo de controle da

¹Também conhecido como princípio de Hollywood: “Não nos chame, nós chamamos você”

aplicação. Assim, as instâncias das classes desenvolvidas esperam ser chamadas pelo framework.

3.3 Framework x Outros Conceitos

Embora haja similaridade entre frameworks e outros conceitos da orientação a objetos, é importante uma explicação mostrando a diferença entre eles. Segundo [Mattsson 1996] alguns desses conceitos são :

- *Bibliotecas de Classes*: As classes em uma biblioteca de classes não são relacionadas frequentemente a um domínio de aplicação específico, diferentemente do framework, que possui um conjunto de classes relacionadas, a fim de prover a funcionalidade de um propósito geral. Algumas diferenças podem ser visualizadas na tabela 3.1

Bibliotecas de Classe	Framework
Conjunto de classes instanciadas pelo cliente	Provê customização pela subclasse
Cientes chamam funções	Chamam funções do cliente
Sem controle de fluxo pré-definido	Controla o fluxo de execução
Sem interação pré-definida	Define a interação do objeto
Sem implementação padrão	Prove comportamento padrão

Tabela 3.1: Bibliotecas de Classe X Framework [Taligent 1994]

- *Padrão de Projetos*: Um padrão de projeto orientado a objetos, segundo Gamma, se difere de um framework em três modos [Gamma et al. 1995]:
 - Os padrões de projeto são mais abstratos que um framework. Frameworks são incorporados no código. No caso dos padrões de projeto, somente exemplos desses são incorporados no código. Os padrões de projetos também descrevem qual é o objetivo, o acordo, e as consequências de um projeto.

- Padrões de projetos são arquiteturas menores que os frameworks. Um framework pode conter mais de um padrão de projeto, como por exemplo, usar o padrão *Factory Method* que define uma interface para criar um objeto e o padrão *Decorator* que é responsável por adicionar responsabilidades dinamicamente a um objeto. Entretanto, o contrário não é possível.
- Frameworks são mais especializados do que padrões de projeto e são sempre descritos com uma aplicação de domínio específico, enquanto que padrões de projeto são comuns e podem ser aplicados em qualquer domínio de aplicação.
- *Padrão de Linguagem*: Um padrão de linguagem descreve como fazer um projeto, enquanto que um framework orientado a objetos é um projeto. Vale a pena ressaltar que, Padrões de Linguagens complementam um framework, descrevendo o porque ele foi projetado dessa maneira.
- *Aplicação orientada a objetos*: Uma aplicação orientada a objetos difere de um framework a medida que a aplicação descreve um programa executável completo que satisfaça uma especificação requerida, por exemplo, análise, projeto e implementação. Um framework em contraste captura as funcionalidades da aplicação, mas não é necessariamente executável porque não cobre o comportamento de uma aplicação específica.

A diferença no processo do desenvolvimento é mostrada na Figura 3.1.

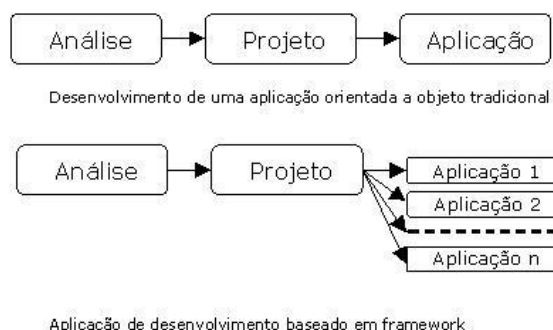


Figura 3.1: Aplicação Baseada em Framework [Mattsson 1996]

3.4 Benefícios dos Frameworks

Os frameworks orientado a objetos trazem benefícios consideráveis para o desenvolvimento de aplicações. Porém, para que esses sejam obtidos o framework deve ser bem projetado, tornando-se mais simples de adicionar extensões fora das funcionalidades comuns do mesmo, possibilitando a interoperabilidade, melhorando a manutenção e a confiabilidade do software. Para [Fayad e Schmidt 1997], os primeiros benefícios obtidos no uso de um framework orientado a objetos são derivados de:

- *Modularidade*: Provê o encapsulamento de detalhes de implementações, melhorando a qualidade do software e reduzindo o esforço requerido em compreender e manter aplicações existentes. Poupano o desenvolvedor de detalhes que não seriam necessários.
- *Reusabilidade*: A característica fundamental que cerca o conceito de framework é o reuso, pois diminui o esforço dos desenvolvedores, evita a recriação e revalidação de soluções comuns às exigências da aplicação;
- *Extensibilidade*: Permite ao desenvolvedor adicionar, substituir e modificar funcionalidades com maior facilidade, customizando as características e serviços de uma nova aplicação.

Segundo a empresa [Taligent 1994], além das características citadas acima, um framework deve ser:

- *Flexível*. Para que suas abstrações possam ser usadas em contextos diferentes. Geralmente a flexibilidade deve ser introduzida dentro do framework em doses apropriadas, estes pontos são chamados de *hot-spots*² e estão diretamente relacionados com a qualidade do framework [Pree 1997].
- *Compreensível*. A interação dos desenvolvedores com o framework deve ser clara, estando bem documentado e fornecendo aplicações como exemplos para o uso dele.

²Os Hot-Spots são pontos de refinamento pré-definidos no framework que serão especializados [Pree 2000]

Pois, se os desenvolvedores do framework e seus usuários(desenvolvedores das aplicações) usarem uma implementação padrão o entendimento será facilitado.

Entretanto, para que um framework possa proporcionar os benefícios desejados, deve-se verificar o custo dessas aplicações. Através da Figura 3.2, pode-se observar que o custo no projeto de um framework esta diretamente relacionado com os benefícios do mesmo. Por exemplo, os benefícios em reusar um framework não aparecem nas primeiras aplicações, e sim, no uso múltiplo dessa tecnologia [Taligent 1995], fazendo com que a relação custo/benefício oscile conforme seu uso.



Figura 3.2: Relação custos x benefícios [Taligent 1994]

3.5 Vantagens e Desvantagens

Na seção anterior foram demonstrados os benefícios no desenvolvimento e uso de frameworks. Nesta seção serão abordadas as vantagens e desvantagens que tem implicações oriundas dos benefícios descritos anteriormente.

Uma das principais vantagens de se utilizar um framework bem planejado advém da possibilidade de redução do custo no desenvolvimento da aplicação, devido

ao alto nível de reuso de análise, projeto e código. Além disso, não requer uma nova tecnologia, podendo ser implementado com qualquer linguagem de programação orientada a objetos já existente [Roberts e Johnson 1996]. O uso de frameworks provoca um impacto positivo na qualidade do software, pois seus componentes já foram previamente testados [Pree 1997]. Com isso, a diminuição de erros nas aplicações se torna significativa.

Existem algumas desvantagens no desenvolvimento de um framework. Segundo [Roberts e Johnson 1996], desenvolver um bom framework requer um alto custo e tem um grau de complexidade elevado, pois exige um planejamento eficaz para garantir flexibilidade em relação a mudanças futuras. Existe ainda a dificuldade no uso desses frameworks, pois, como são modelos reutilizáveis abstratos, precisam de desenvolvedores experientes para especializar e concretizar seu uso.

Um exemplo citado por [Silva 2004] esclarece as vantagens e desvantagens de se utilizar um framework orientado a objetos:

“Um framework é como se fosse um formulário. O formulário já vem pronto, o que facilita seu preenchimento, entretanto, a flexibilidade de como e onde escrever se limita as lacunas oferecidas pelo documento. O oposto disso, seria o preenchimento de um papel em branco, a desvantagem seria de escrever o documento em sua totalidade, porém, a flexibilidade na escolha do estilo e local da escrita é livre.”

3.6 Frameworks Caixa-Branca e Caixa-Preta

Os frameworks são constituídos de uma estrutura básica, na qual classes podem ser adicionadas, removidas ou estendidas de acordo com a necessidade de uma determinada aplicação. Esses podem ser classificados à partir das técnicas utilizadas para estendê-los, conhecidas como caixa branca e caixa preta.

O desenvolvimento de um framework geralmente é iniciado com uma abordagem caixa-branca, o que deve ser visto como um estágio natural da evolução de um sistema [Hautamaki 1997]. Entretanto, segundo [Johnson e Foote 1988], os frameworks caixa-branca tendem a evoluir para caixa-preta.

Os frameworks caixa-branca (também chamados de dirigido a arquitetura) são baseados em herança e suas funcionalidades são estendidas herdando das classes do framework, os métodos pré-definidos nele [Fayad e Schmidt 1997]. Os frameworks caixa-branca podem ser difíceis de usar porque requerem um maior esforço no desenvolvimento de códigos, pois, as subclasses das aplicações são desenvolvidas com base nas classes pré-definidas no framework.

Os frameworks caixa-preta (também chamados de dirigido a dados) são baseados em composição. Assim, novas funcionalidades são obtidas montando ou compondo objetos definidos pelo desenvolvedor da aplicação. Segundo a empresa [Taligent 1994], esses são mais simples de usar, porém possuem menor flexibilidade.

Uma terceira técnica de uso de frameworks é a união dos dois conceitos citados acima. Com isto, um framework caixa-cinza terá maior flexibilidade ao usar o conceito de herança proposto pelo framework caixa-branca, mas capacidade de extensão por customização e facilidades de uso encontrados no framework caixa-preta.

3.7 Desenvolvendo Framework

O processo de desenvolvimento de um framework depende da experiências dos envolvidos no projeto sobre o domínio do problema a ser tratado neste framework [Mattsson 1996]. Existem alguns processos para o desenvolvimento de um framework entretanto, neste trabalho será apresentado o processo de desenvolvimento baseado em domínio.

O primeiro passo para analisar o domínio do problema a ser tratado no framework é identificar as primeiras abstrações, normalmente a análise é feita através de algumas aplicações existentes.

Segundo [Mattsson 1996], não é uma tarefa fácil analisar o domínio de aplicações existentes, mas, é importante a realização deste processo para que possa ser efetuada as modificações na estrutura do framework quando necessárias. A Figura 3.3 representa este processo de desenvolvimento.

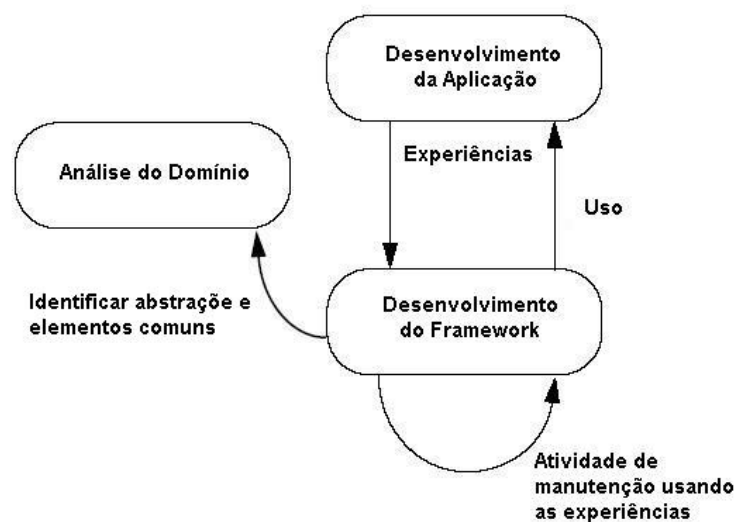


Figura 3.3: Processo do desenvolvimento do framework baseado em análise de domínio
[Mattsson 1996]

[Prieto-Díaz e Arango 1991] define a análise de domínio como um processo de identificação e organização de conhecimento a respeito de uma classe de problemas – um domínio de aplicações – para suportar a descrição e solução destes problemas.

Para [Silva 2000], “o objetivo da *Análise de Domínio* é produzir subsídios para a reutilização de software. Estes subsídios correspondem a uma descrição do domínio contida em um modelo do domínio”.

A análise do domínio pode ser vista como uma análise mais ampla e mais extensiva que tente capturar as exigências do domínio do problema incluindo as exigências futuras. A Figura 3.4 representa a generalização de um domínio a partir da análise de algumas aplicações. A intersecção das aplicações para abstração das características (estruturas e funcionalidades) comuns entre elas é essencial, pois, são destas características que será gerado o projeto do framework.

Segundo [Silva 2000], conhecer um conjunto de domínios para geração de um framework demanda muito esforço, além de não caracterizar uma interação amigável com o usuário. Sendo assim, a empresa [Taligent 1994] cita alguns passos além da

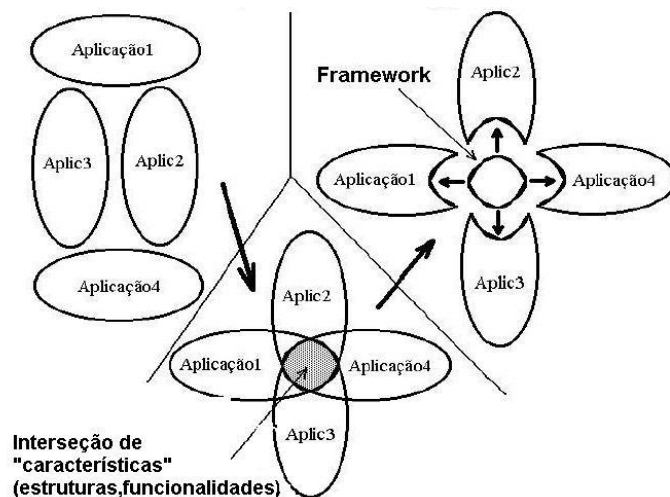


Figura 3.4: Generalização do Domínio [Silva 2003]

abstração do domínio como importantes para o sucesso do projeto como: projetar a interação dos clientes com o framework, implementar, testar e refinar o projeto.

3.8 Exemplos de Framework Orientado a Objetos

Nesta seção serão apresentados dois exemplos de framework orientado a objetos. O Framework FraG é um framework para jogos de tabuleiro e o Framework HyperToolBuilder para a criação de ferramentas de autoria para documentos multimídia.

3.8.1 Framework FraG

O FraG é um framework orientado a objetos que foi desenvolvido por [Silva 2000] para geração de jogos de tabuleiro. O que caracteriza seu domínio são os diferentes jogos de tabuleiro que podem ser desenvolvidos a partir dele, como jogo da velha, dama, gamão, dentre outros.

O framework FraG passou por algumas etapas ao longo do seu desenvolvimento: a generalização, onde foram identificadas as estruturas idênticas das aplicações analisadas; a flexibilização, que foram identificados os hot-spots para manter a flexibili-

dade da estrutura; a de aplicações de princípios práticos de orientação a objetos, onde foi mantida a flexibilidade de uso através da herança e composição de objetos, dentre outras.

Através do framework FraG foram desenvolvidas três aplicações: o Jogo da Velha, o Jogo Corrida e o Jogo Banco Imobiliário. Vale a pena ressaltar que, no projeto do framework foram definidos que toda aplicação deve possuir uma subclasse concreta de `GameInterface` e uma de `Board`, que são classes abstratas, redefiníveis e essenciais³ [Silva 2000]. Na Figura 3.5 é apresentado o diagrama de classes do framework FraG com as duas classes que foram estendidas para o desenvolvimento da aplicação Jogo da Velha.

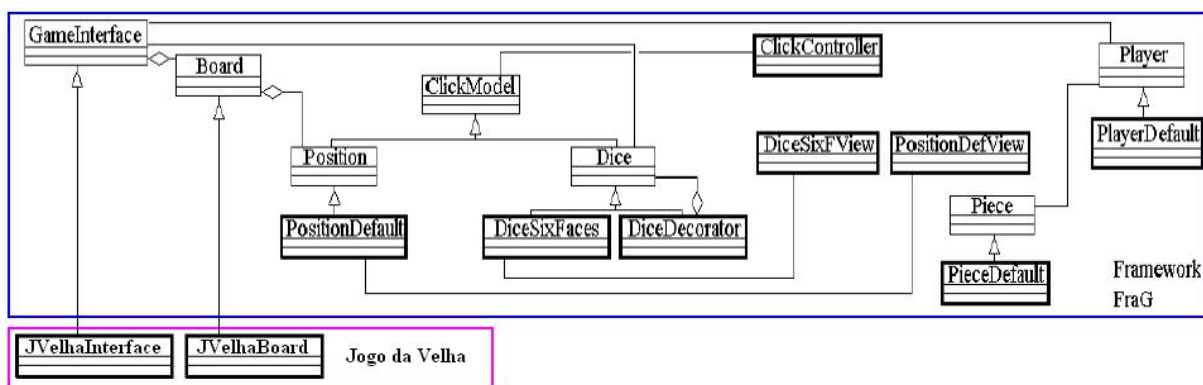


Figura 3.5: Aplicação Jogo da Velha [Silva 2000]

Todas as funcionalidades comuns às aplicações citadas acima estão implementadas no framework e as classes estendidas apresentam as características particulares da aplicação Jogo da Velha. A classe `JVelhaInterface` representa a interface com o usuário, enquanto a classe `JVelhaBoard` a dinâmica do jogo. Na Figura 3.6 é apresentada a interface da aplicação do Jogo da Velha.

³Os conceitos de classes redefiníveis e essenciais serão discutido no capítulo 5

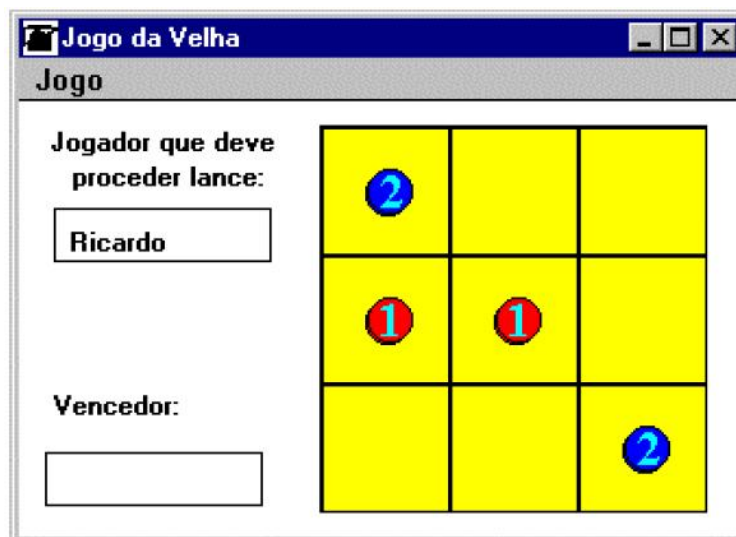


Figura 3.6: Aplicação sob FraG [Silva 2000]

3.8.2 Framework HyperToolBuilder

O HyperToolBuilder é um framework orientado a objetos, projetado por [Beutler 2003] para facilitar o desenvolvimento de ferramentas de autoria⁴ para a criação e edição de documentos multimídia. Ele provê funcionalidades para a construção do ambiente de autoria, a estruturação dos documentos, a persistência e a execução desses documentos.

O objetivo do HyperToolBuilder é fornecer suporte de reutilização em alta granularidade, tanto de código como de projeto de software para a produção dessas aplicações. Na Figura 3.8 é apresentado o diagrama de classes da estrutura do framework.

Segundo [Beutler 2003], toda a aplicação desenvolvida sob o framework HyperToolBuilder deverá especializar a classe *AuthoringTool*. Esta é responsável por centralizar todo o controle da ferramenta de autoria e agregar algumas classes tais como: a classe *MenuBar* que é responsável por receber e executar os comandos disponíveis pela aplicação; a classe *ToolPanel* que corresponde ao painel que contém as ferramentas para

⁴O termo ferramenta de autoria refere-se a aplicações gerais para criação de qualquer documento multimídia, em qualquer área do conhecimento [Beutler 2003].

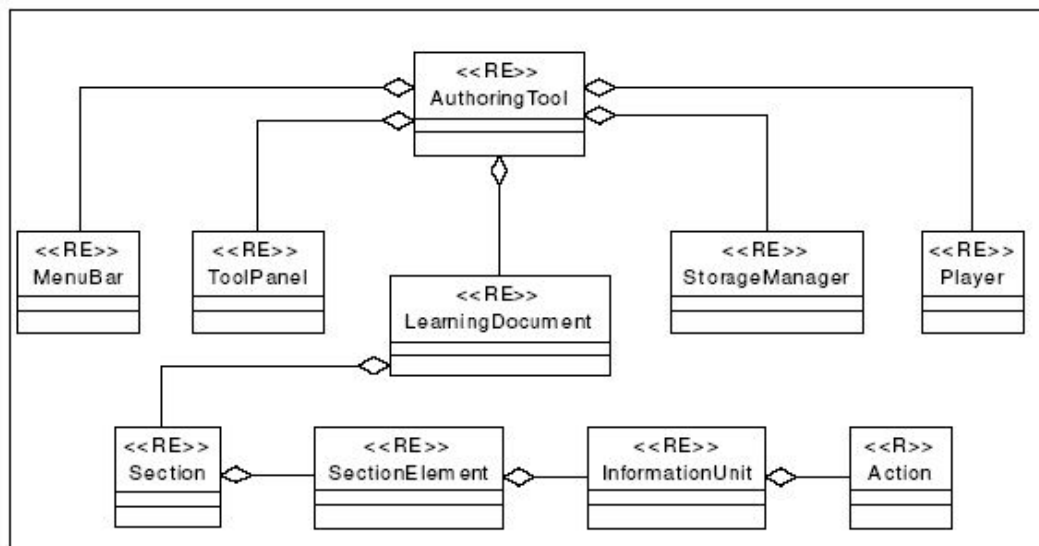


Figura 3.7: Diagrama da estrutura do framework HyperToolBuilder [Beutler 2003]

cada aplicação; a classe Learning Document, que representa o documento de ensino; a classe StorageManager que possui os recursos necessários para persistência dos documentos produzidos e a classe Player que é responsável pela execução do documento de ensino.

Através do framework HyperToolBuilder foram desenvolvidas duas aplicações: O HyperBook, que é uma ferramenta de autoria para livros eletrônicos e o HyperTest, que é ferramenta de autoria para construção de exercícios, entretanto, neste trabalho será apresentado resumidamente a ferramenta HyperBook. O objetivo desta ferramenta é possibilitar a criação e visualização de livros eletrônicos, o usuário que irá desenvolver o livro poderá utilizar todas as ferramentas disponíveis no sistema. Após o término todos os usuário poderão visualizar o livro.

Para a criação da ferramenta de autoria HyperBook foi desenvolvida uma subclasse concreta BookAuthoringTool, da classe abstrata AuthoringTool do framework, fazendo com que a aplicação herde os relacionamentos das suas superclasses no framework [Beutler 2003]. Na Figura 3.8, é apresentada uma página de uma mídia após a definição das propriedades do HyperBook.



Figura 3.8: HyperBook [Beutler 2003]

3.9 Conclusão

A principal vantagem do framework orientado a objetos é a reusabilidade oferecida por ele. Com esta característica há uma diminuição na quantidade de código a ser desenvolvido e conseqüentemente uma menor probabilidade de erro.

Portanto um dos maiores desafios em se utilizar um framework é a curva de aprendizagem, para que o aprendizado do framework seja facilitado, este deve prover uma boa documentação. Vale a pena ressaltar que, após a obtenção de conhecimento do framework, as aplicações podem ser geradas de uma maneira mais rápida, menos custosa e mais confiável. Entretanto, a necessidade de um domínio bem definido é essencial para o sucesso do framework.

Sendo assim, este capítulo apresentou algumas características de framework orientado a objetos úteis à compreensão do capítulo 4. No qual serão apresentados alguns frameworks para ambientes colaborativos.

Capítulo 4

Frameworks para Ambientes Colaborativos

4.1 Introdução

Os ambientes de aprendizagem colaborativa apoiada por computador ou CSCL, surgiram a partir dos ambientes de CSCW (Computer Supported Cooperative Work) que também trabalham com colaboração em grupos. Entretanto, o enfoque do CSCW não é direcionado para educação. Uma definição encontrada no dicionário computacional se refere ao CSCW como *“uma ferramenta de software e tecnologia que suporta grupos de pessoas trabalhando juntas em um projeto, frequentemente em locais diferentes”*.

A diferença básica entre CSCL e CSCW citado por [Hsiao 2001] é que o conceito de *“CSCW é usado principalmente no cenário de negócios e sua finalidade é facilitar uma comunicação e a produtividade do grupo, enquanto que CSCL é usado no cenário educacional e sua finalidade é auxiliar os alunos no aprendizado colaborativo”*. Ele ainda acrescenta que ambos são baseados na promessa que os sistemas apoiados computador podem suportar e facilitar o processo do grupo. Porém, não são projetados para substituir uma comunicação face a face.

4.2 O Framework Habanero

O NCSA Habanero foi desenvolvido pela *National Center for Supercomputing Applications*. Este framework foi projetado para fornecer aos desenvolvedores ferramentas necessárias para criação de aplicações colaborativas em Java, fornecendo os métodos necessários que tornam possível a criação ou extensão dessas aplicações.

Usando o framework Habanero pode-se criar e trabalhar em aplicações compartilhadas a partir de locais remotos na internet. Permite também aos desenvolvedores de *groupware* construírem aplicações melhores em menor tempo, incluindo todas as facilidades do trabalho em rede bem como a serialização, a abstração, e a distribuição de software necessários para compartilhar o estado dos dados e os eventos chave entre os múltiplos colaboradores. Não há nenhum limite inerente ao número das ferramentas por a sessão, nem há um limite no tipo de ferramentas que podem ser compartilhadas [Habanero 1996]. Na Figura 4.1 três ferramentas desenvolvidas sobre o framework Habanero serão apresentadas.

- A ferramenta *Chat* que usa o mecanismo de compartilhamento de objetos e tem habilidade de registrar uma sessão do bate-papo para a referência futura.
- A ferramenta *Whiteboard*, usada para compartilhar imagens e desenhos durante sua colaboração.
- A ferramenta *Voting*, traz a democracia a uma sessão colaborativa, no qual um membro define a pergunta e a modalidade da resposta, envia para o grupo e recebe uma janela do voto. Depois que o voto está completo, cada membro recebe uma janela do resultado que indica a contagem do voto.

A partir destas ferramentas podem ser desenvolvidos qualquer tipo de ambiente colaborativo. Além disso, todas as ferramentas desenvolvidas e toda programação Java que compõe o framework Habanero estão disponíveis na internet.

Embora o enfoque no desenvolvimento do framework Habanero não tenha sido voltado para prover suporte pedagógico, as ferramentas criadas sob ele podem

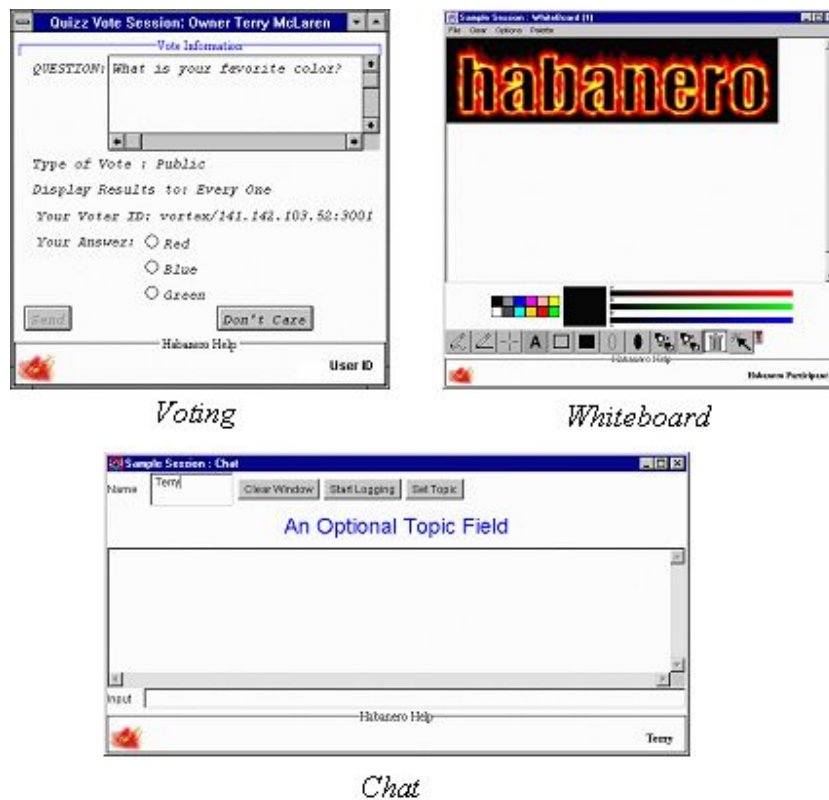


Figura 4.1: Ferramentas do Habanero [Habanero 1996]

ser facilmente utilizadas como ferramentas didáticas em ambientes de aprendizagem colaborativa.

4.2.1 O Ambiente Habanero

O ambiente do Habanero fornece os requisitos necessários para criar aplicações de trabalho colaborativo e comunidades virtuais [Chabert, Jackson e Pietrovicz 1997]. Na figura 4.2 é apresentado um ambiente colaborativo que mostra dois médicos interagindo, compartilhando seus conhecimentos e diagnósticos de seus pacientes.

A comunicação entre eles é feita através das ferramentas desenvolvidas sob o framework Habanero. A ferramenta de *Chat* permite aos médicos visualizar os dados que serão compartilhados. Usando as ferramentas *Whiteboard* e *Voting* os médicos

podem chegar a um diagnóstico comum por meio de experiências e opiniões diferentes.



Figura 4.2: Ambiente Halbanero [Chabert 1997]

Segundo [Santos 1999], o Habanero é um framework que pode apoiar formas inovadoras de aprendizagem, porém essas potencialidades não são exploradas pelos professores da educação básica (público a quem, o Habanero é dirigido) por falta de suporte pedagógico.

4.3 O Framework Promondia

O Promondia ¹ é um framework orientado a objetos que tem como objetivo prover a comunicação de grupos em tempo real pela Internet. Ele foi desenvolvido utilizando a linguagem de programação Java e pode ser executado em qualquer sistema operacional que tenha um navegador que suporte Java [Technology e Erlangen 1997].

O framework Promondia pode ser estendido para suportar novos métodos

¹Sua versão anterior era conhecida como COMO.

de comunicação e também o desenvolvimento das aplicações tais como *whiteboards* compartilhados, *chat*, entre outras. Segundo [Santos 1999], estas ferramentas são interativas, o que faz este sistema potencialmente útil para aprendizagem cooperativa. Na figura 4.4 será apresentado um exemplo da ferramenta de *chat* baseado em texto.

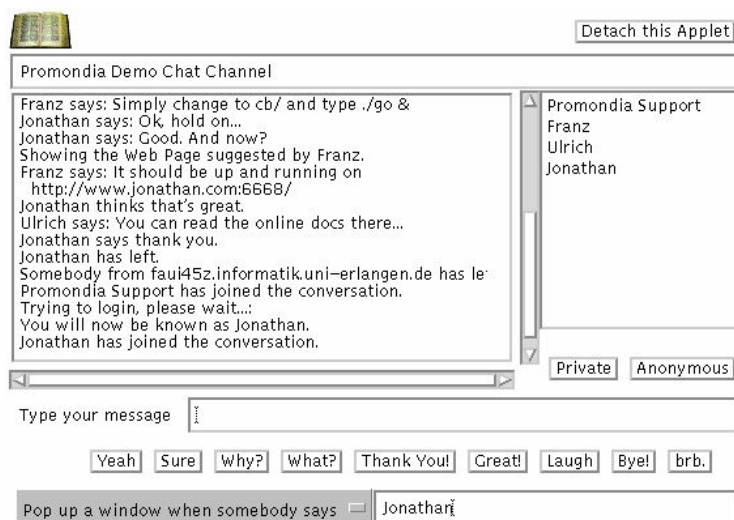


Figura 4.3: Chat [Gall e Hauck 1997]

A ferramenta *chat* oferece uma variedade de módulos, desde os mais simples até os mais sofisticados, permitindo mensagens privadas e anônimas, notificações sobre certos eventos e mensagens de audio. Os usuários podem convidar outras pessoas para participarem de um bate-papo, no qual um deles, o cliente moderador, permite ao administrador, outro usuário, controlar a conversação do bate-papo, por meio da definição de diferentes direitos (fala livre, moderada, mudo)[Gall e Hauck 1997].

Outra ferramenta desenvolvida utilizando o framework Promondia é a *whiteboard* compartilhado. Ela é uma Applet Java para desenho vetorial multi-usuário, na qual podem ser enviadas imagens e textos. Ainda, tal ferramenta permite ao usuário desenhar sobre um *Whiteboard* junto com qualquer pessoa que esteja atualmente visitando a página que contém essa Applet. Na figura 4.4 é apresentado um exemplo desta ferramenta.

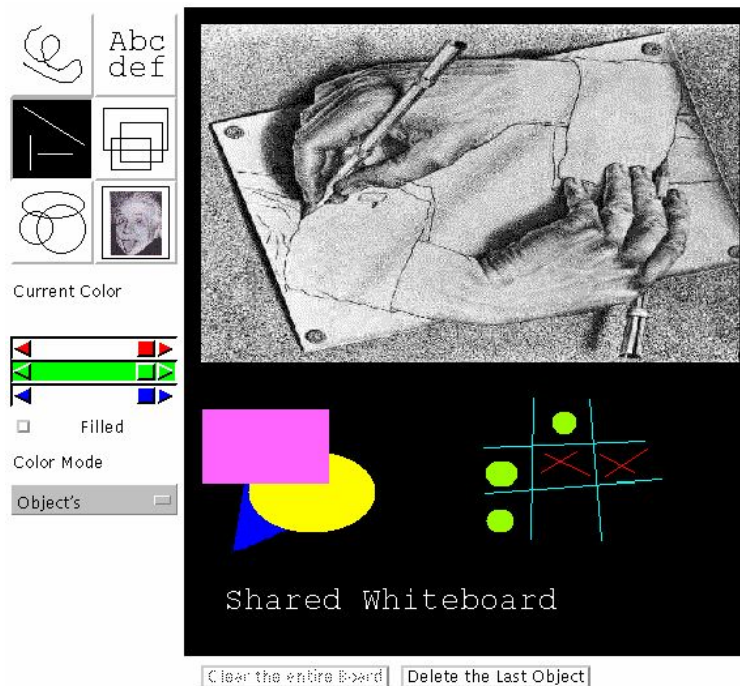


Figura 4.4: *Whiteboard* Compartilhado [Gall e Hauck 1997]

Um gerenciador de sessões, prove informação sobre quais sessões estão rodando e quais usuários estão disponíveis, usuário comum ou administrador. Um usuário comum pode selecionar sessões e participar delas, pode também convidar outros usuários por email. Já o administrador pode mudar as propriedades e configurações das sessões e desconectar usuários de uma sessão [Gall e Hauck 1997].

4.4 O Framework DARE

O framework DARE(Activités Distribuées dans un Environnement Réflexif), foi desenvolvido por [Bourguin 2000] com o objetivo de integrar os conceitos da atividade humana, tendo como base a Teoria da Atividade(TA) aos fundamentos de CSCW. A forma utilizada para integrar estes conceitos a definição dos princípios da TA dentro de um ambiente CSCW.

A base do framework DARE está diretamente ligada aos princípios de-

finidos da TA proposta por Engeström. Assim, os conceitos da atividade foram definidos da seguinte forma:

- na *ferramenta* é definido tudo o que será utilizado no processo de transformação e mais as ferramentas materiais.
- as *regras* são definidas como um conjunto leis praticas aceitas dentro da comunidade, podendo ser implícitas ou explícitas.
- o *objeto* é o motivo da atividade
- o *sujeito* tem papel fundamental, não podendo haver transformação do objeto se não tiver um sujeito que trabalhe.
- a *comunidade* é um conjunto de sujeitos que partilham o mesmo objeto na atividade.
- a *divisão de trabalho* é a organização explícita ou implícita da comunidade.

A Figura 4.5 apresenta o modelo do framework DARE.



Figura 4.5: Estrutura de uma atividade no DARE [Bourguin 2000]

O papel é representado pela divisão de trabalho e corresponde ao sujeito dentro da comunidade, definindo o que um sujeito deve fazer e o que ele é. Um sujeito executa um certo papel realizando uma parte do trabalho. No modelo DARE esse papel

inclui os conceitos de regras e divisão de trabalho determinando o que é ser um membro da comunidade.

Este modelo representa uma síntese da estrutura da atividade proposta por Engeström, que segundo [Bourguin 2000] não nega os princípios da TA. No qual o papel passa a ser artefato mediador da atividade.

A figura 4.6, representa as principais interrelações existentes entre os conceitos definidos na estrutura básica da TA e os pontos introduzidos para conceber o modelo conceitual do DARE.

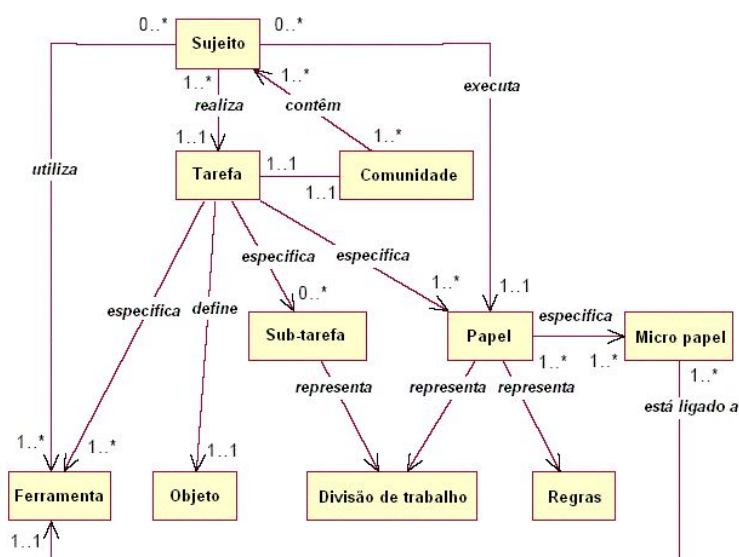


Figura 4.6: Os conceitos da estrutura básica de uma atividade e o modelo conceitual do DARE [Bourguin 2000]

A comunidade contém sujeitos que realizam tarefas, utilizam ferramentas e executam um papel dentro da comunidade. Um papel é constituído por vários micro-papeis que descreve os direitos e deveres dos sujeitos. No DARE cada papel é um componente que pode ser usado e modificado. O papel representa as regras e divisão do trabalho.

Uma tarefa define um conjunto de papeis. Especifica as sub-tarefas que devem ser realizadas e define um objeto. A tarefa também define um conjunto de ferramentas que permite aos sujeitos realizar seu trabalho em função de seus papeis. As

ferramentas estão ligas aos micro-papeis assumidos pelo sujeito.

4.4.1 O Ambiente DARE

A partir dos conceitos definidos no framework DARE, foi desenvolvido um ambiente reflexível que suporta um conjunto de atividades cooperativas baseadas na TA. Esse ambiente gera um conjunto de ferramentas e papeis seguindo os princípios definidos no framework. No ambiente DARE as ferramentas são componentes de software (applets Java) desenvolvidos fora do ambiente. Elas são integradas na atividade, como por exemplo, ferramenta de chat, editor de audio, dentre outras. A Figura 4.7 apresenta a tela Atividade.



Figura 4.7: Exemplo de atividade [Bourguin 2000]

Esta é uma das applets desenvolvidas na implementação do DARE. No qual um *sujeito* identificado como Greg, visualiza o conjunto de *ferramentas* que são acessíveis para o *papel* dele na atividade (Professor), de forma semelhante a lista de membros que são esperados ou que já chegaram. Um ponto a ser destacado é a utilização das ferramentas pelos sujeitos em função do seu papel. Pois, o papel de um sujeito, através dos seus micro-papéis, determina as ações que ele pode executar.

Na tela Ferramenta, figura 4.8, um sujeito pode: criar ou alterar uma ferramenta necessária para a execução de sua atividade, definir as informações necessárias para a associação de um componente externo (URL, parâmetros, etc.) a uma ferramenta, listar as operações definidas para esta ferramenta, além da possibilidade de criar novas operações, alterá-las ou ainda suprimi-las. Nesta tela foi acrescentado, ainda, uma lista dos micro-papeis associados a ferramenta.

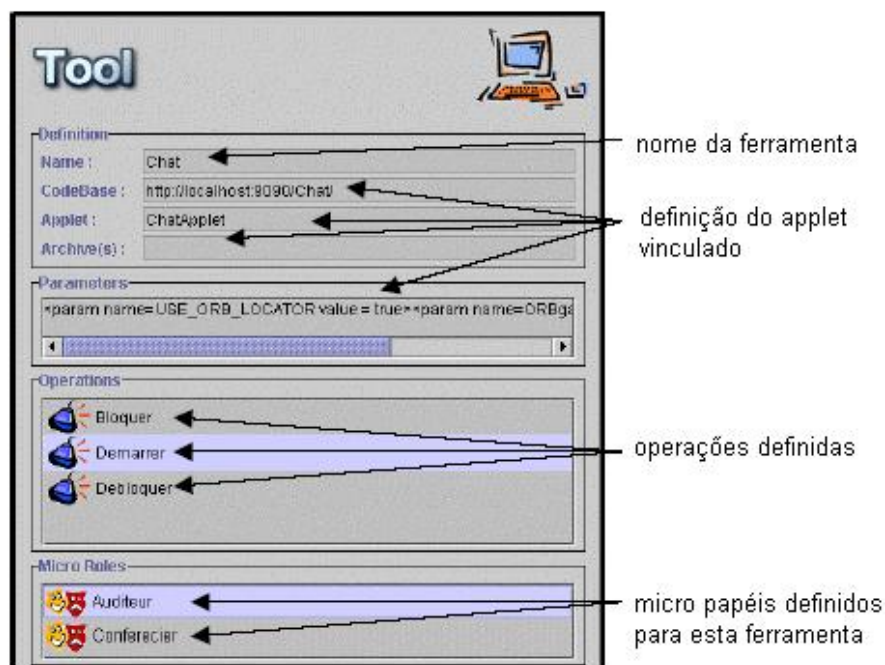


Figura 4.8: Exemplo da ferramenta [Bourguin 2000]

Na figura 4.9 apresenta-se a tela Tarefa. Nela acrescentar, retirar ou alterar são opções disponíveis para as ferramentas de uma tarefa. O sujeito pode editar, adicionar ou remover as ferramentas vinculadas à tarefa desta tela e os papéis disponíveis para cada tarefa. Cada conceitos da TA tem uma característica particular no DARE.

- o objeto não é um elemento isolado, mas sim um atributo da classe tarefa.
- na classe tarefa o sujeito recebe um papel que vai determinar quais as ferramentas que ele poderá utilizar dentro daquela tarefa.

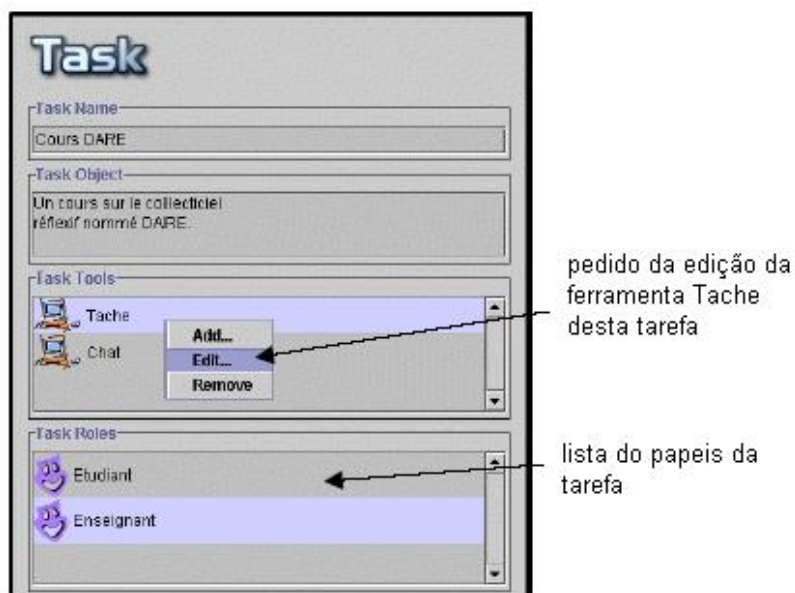


Figura 4.9: Exemplo da tarefa [Bourguin 2000]

- um conjunto de sujeitos que trabalham o mesmo objeto em uma atividade é denominada comunidade. Em uma atividade só pode existir uma comunidade. Esta representa na tarefa somente um conjunto de papeis que são especificados e que podem ser executados pelos sujeitos.
- O nome da ferramenta pode ser modificado de uma atividade para outra e as ferramentas que serão utilizadas dependem do objeto da tarefa realizada. O nome da ferramenta não é limitado e depende da vontade do sujeito.

4.5 Conclusão

Os frameworks Habanero e Promondia apresentados neste capítulo não foram desenvolvidos tendo como foco a preocupação pedagógica, porém as ferramentas desenvolvidas sobre estes podem e são usadas para fins educacionais. Entretanto, o que se observa é uma importância direcionada aos aspectos operacionais das ferramentas.

O framework DARE, foi desenvolvido baseado na Teoria da Atividade, conceito visto no capítulo 2. Pode-se dizer que o DARE se aproxima mais do framework proposto neste trabalho pelo fato de utilizarem a mesma teoria. Um outro ponto de semelhança diz respeito à função dos mediadores na estrutura, como, a divisão de tarefas, a atribuição dos papéis de cada membro e o conceito de comunidade no desenvolvimento de cada atividade.

Porém, vale a pena ressaltar que, o FAAC além de se basear nos conceitos citados, provê uma associação entre a estrutura e os níveis da atividade apresentadas na teoria. Esta associação será descrita no capítulo 5.

Capítulo 5

Um Modelo Computacional para a Teoria da Atividade

5.1 Introdução

No desenvolvimento de softwares educacionais é necessário o envolvimento de profissionais de áreas distintas, como foi apresentado no capítulo 1. Além deste fator, é importante destacar a relevância de uma teoria aplicada à educação como apoio no desenvolvimento destes softwares.

Normalmente o diálogo entre os profissionais envolvidos no desenvolvimento de softwares educacionais é um tanto custoso, pela visão diferente de cada profissional e devido à formação distinta dos mesmos.

A fim de amenizar este entendimento, foi proposto o projeto de um framework orientado a objetos tendo como base uma teoria aplicada à educação. A escolha desta técnica ocorreu pelo fato de ser uma estrutura semi-acabada e reutilizável.

Entretanto, a teoria adotada no contexto dessa dissertação, a Teoria da Atividade, é considerada uma teoria multidisciplinar e com possibilidades para qualquer área de atuação humana, não possuindo uma especificação para área educacional. Com isto, antes de iniciar o projeto do framework, foi criado um modelo computacional para

servir de elo entre a teoria e a técnica de framework. A representação deste modelo pode ser visualizado na Figura 5.1.

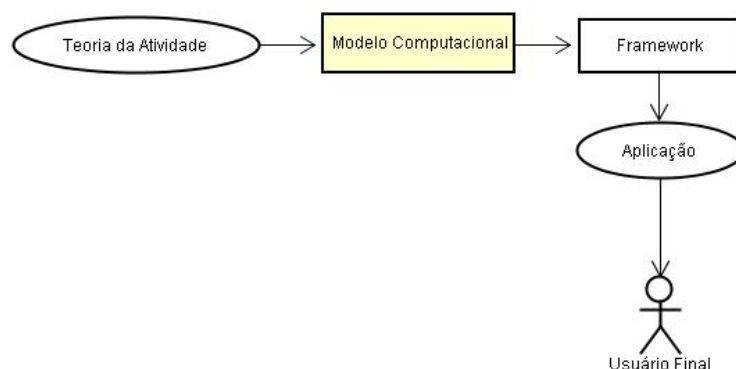


Figura 5.1: Adaptação do Níveis da Atividade

Para representar o Modelo Computacional (MC), foi adicionado aos elementos da Teoria da Atividade atributos e métodos, que são inexistentes nos conceitos da mesma. Os elementos da TA sobre uma visão computacional são apresentados na seção 5.2.

5.2 Elementos da Teoria da Atividade

A estrutura da atividade, apresentada no capítulo 2, é representada por seis elementos, no qual Sujeito, Comunidade e Objeto são chamados de elementos mediados e Ferramenta, Regra e Divisão de Trabalhos de elementos mediadores. Uma característica inerente desta estrutura são as relações que sempre ocorrem a partir de seus mediadores. Como o MC é baseado na TA, esse deve garantir que essas relações aconteçam.

5.2.1 O Mediador Ferramenta

O mediador Ferramenta é representado no MC através da classe *Ferramenta*. Para garantir a mediação entre Sujeito e Objeto é necessário implementar o método *usarFerramenta*, que recebe como parâmetro um objeto da classe *Objeto* e retorna

o mesmo modificado. Cabe ao desenvolvedor da aplicação definir como a ferramenta será implementada e quais alterações serão realizadas no objeto. A Figura 5.2 apresenta a extensão das classes Ferramenta e Objeto.

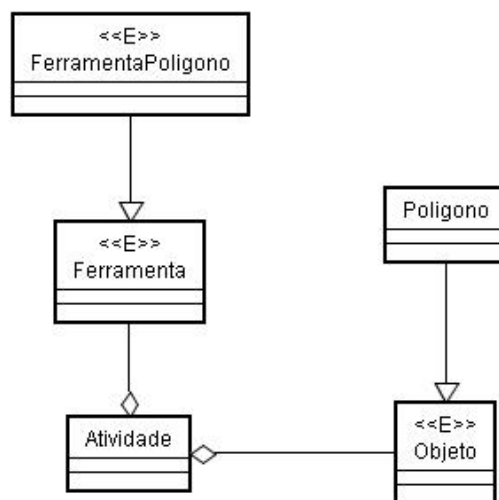


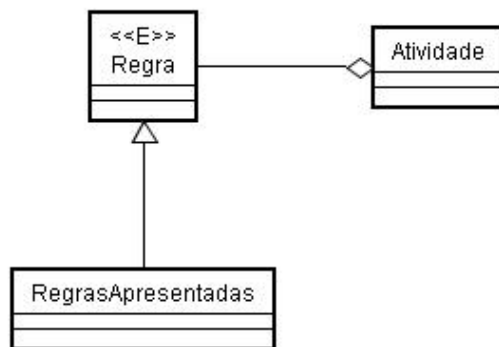
Figura 5.2: Extensão da classe Ferramenta e Objeto

Por exemplo, um software educacional onde o professor deseja apontar as características de um Polígono. O desenvolvedor define o objeto *Polígono*, que deve ser uma subclasse de *Objeto*, com um atributo *numeroLados*. Esse objeto é iniciado com três lados, formando um triângulo. Após a criação do objeto é definida a ferramenta *FerramentaPolígono*, uma subclasse de *Ferramenta*, para transformar o polígono usando o método *usarFerramenta* que pode incrementar ou decrementar o número de lados modificando a forma do objeto.

5.2.2 O Mediador Regra

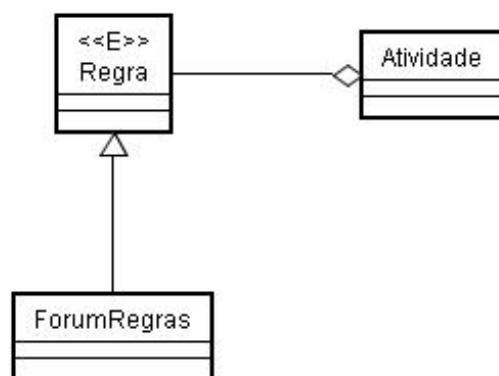
O mediador Regra é representado pela classe *Regra* e deve prover a relação entre Sujeito e Comunidade. Para que haja esta relação é necessária a implementação do método *mostrarRegras*. A classe *RegrasApresentadas*, deve ser uma subclasse de *Regra* e apresenta as regras criadas pela Comunidade. Esta pode ser visualizada na Figura

5.3.

**Figura 5.3:** Extensão da classe Regra

Utilizando o mesmo exemplo dos Polígonos, pode-se dizer que os membros da comunidade somente trabalharam com polígonos de três lados.

Um outro exemplo pode ser a discussão e a criação de novas regras pela comunidade poderão ser realizadas através de um fórum. Este pode ser desenvolvido estendendo a classe *ForumRegras* e implementando alguns novos métodos como por exemplo *mostrarRegrasDefinidas* que exibe aos membros do fórum as regras já definidas. A Figura 5.4 representa esta extensão.

**Figura 5.4:** Cadastro da Atividade

5.2.3 O Mediador Divisão de Trabalho

O mediador Divisão de Trabalho é representado pela classe *Divisao-Trabalho* que media a relação Comunidade e Objeto. Essa deve prover a organização da atividade que será executada, criando papéis para cada membro da comunidade.

A classe *Comunidade* do MC, materializa o conceito de colaboração apresentado na seção 2.6.1 do capítulo 2, pois, ela mostra que uma atividade deve ser desenvolvida por duas ou mais pessoas, o que caracteriza o trabalho em grupo.

Além dos elementos que compõe a atividade, os níveis da TA também são modelados na visão computacional para serem inserido no framework proposto. Estes são apresentados na seção 5.3.

5.3 Níveis da Teoria da Atividade

A Teoria da Atividade como foi apresentada no capítulo 2, é composta por três níveis: a própria Atividade, a Ação e a Operação.

A relação entre esses níveis ocorre de uma maneira dinâmica podendo oscilar de ação para operação e voltar ao nível de atividade. Assim, como a ação é considerada o planejamento de como será realizada uma atividade, quando o sujeito não precisa mais pensar em como realiza-la esta estará no nível de operação e quando um fato desconhecido ocorrer, esta retorna novamente ao nível de atividade. Este dinamismo entre os níveis ocorre até o fim da atividade inteira. Pode-se dizer que, em uma atividade “Dirigir um Carro”, o sujeito passa pelos três níveis até que esta esteja internalizada nele.

O nível Atividade é representado no MC pela classe *Tarefa*. Esta atua como uma parte da atividade principal que é criada pela super classe *DivisaoTrabalho*. O atributo *papel* na classe *Tarefa* representa o que cada sujeito pode realizar dentro desta.

A Ação é representada pela classe *Acao*, que é uma subclasse de *Tarefa*. A classe *Acao* centraliza as possíveis ações de cada membro da comunidade. Estas ações se referem ao planejamento consciente de como será executada uma atividade.

A execução de uma ação depende diretamente das operações, que é descrita no MC na classe *Operacao*. Esta apresenta a parte operacional da atividade, que ocorre a partir das condições para sua realização. Assim, a classe *Operação* é responsável por listar as ferramentas disponíveis na atividade para a realização da mesma. Isto ocorre com a implementação do método *listarFerramenta*.

No exemplo do Polígono, citado acima, uma Tarefa pode ser mostrar as características do polígono. Na Ação, este exemplo pode ser o "como" estas características serão estudadas, e por fim, a Operação é maneira prática da realização da ação.

5.4 Associação da Estrutura e dos Níveis de Atividade

Na visão da TA, uma atividade é composta pelos elementos que estão representados no triângulo de Engeström. Porém, outro conceito que compõe esta teoria, mas não está explícita no triângulo, são os níveis de uma atividade. Para Leont'ev, uma atividade poder possuir varias ações e operações.

Como pode ser observado na Figura 5.5, a estrutura da atividade é composta por três relacionamentos onde o Sujeito e a Comunidade são mediados por Regras, o Sujeito e o Objeto mediados pela Ferramenta e a Comunidade e o Objeto mediados pela Divisão de Trabalho e o nível é composto pela própria Atividade, pela Ação e pela Operação.



Figura 5.5: Triângulo da Atividade

O triângulo representa uma atividade estática, onde, um sujeito se insere

em uma comunidade, divide as tarefas a serem realizadas e se utilizam de ferramentas para alcançarem seu objetivo, que é termino da atividade. Entretanto, para que esta atividade seja concluída e internalizada o sujeito passa pelos os níveis que a compõe. Sendo assim, a partir da divisão de trabalho, foi realizada uma associação dos níveis da atividade. Esta associação, feita pela autora, pode ser observada na Figura 5.6.



Figura 5.6: Associação da Estrutura e dos Níveis da Atividade

No MC, a classe *DivisãoTrabalho* é responsável por atribuir o papel e as tarefas de cada sujeito, as tarefas no modelo computacional representam o primeiro elemento dos níveis que é a própria atividade. A partir desta atribuição, o sujeito começa o planejamento para a realização da sua tarefa, ou seja, neste momento ele se encontra no nível de ação. A operação, como descrito no capítulo 2, é a materialização destas ações, quais as ferramentas que estão disponível para sua realização. Porém, a relação entre os níveis é dinâmica, podendo haver oscilação entre eles.

O MC foi projetado visando o desenvolvimento de frameworks para ambientes de aprendizagem colaborativa. Este conceito é abordado seção 5.5, apresentando algumas possibilidades de utilização do modelo computacional.

5.5 Conceitos de CSCL no MC

Os ambientes de CSCL (Computer Supported Collaborative Learning), conceito abordado no capítulo 4, ressaltam o uso do computador como mediador no pro-

cesso de aprendizagem, o que vem de encontro com a proposta da Teoria da Atividade adaptada por Engeström. Além da mediação, outros conceitos dessa teoria podem ser utilizados em ambientes de CSCL, como: a comunidade, que garante que haja colaboração entre os seus membros; a divisão de trabalho e as regras que organizam a maneira que será desenvolvida a atividade, e por fim, o conceito de ferramenta, que proporciona o desenvolvimento de artefatos para a transformação do objeto da atividade.

Através de uma análise sobre as possibilidades de uso do MC no desenvolvimento do framework proposto, concluiu-se que as aplicações que serão derivadas dele, cada qual com suas particularidades, podem ser estendidas utilizando também os conceitos de CSCL. Pode-se dizer então que o framework proposto além de suas bases serem focadas na TA também utiliza alguns conceitos de CSCL.

Além dos conceitos de aprendizagem colaborativa apresentados neste capítulo, o MC tem possibilidades de adaptar características de ambientes educacionais baseados na TA que são descritos na seção 5.6.

5.6 Implicações Educacionais no MC

Na seção 2.6.1 do capítulo 2 foram apresentados alguns softwares baseados na TA. Nesta seção será descrito como estes princípios podem aplicados no MC.

As atividades autênticas utilizadas nos softwares educacionais Média Fusion e Dinosaur Canyon podem ser identificadas na mediação com a comunidade, pois esta pode ser composta por especialistas e professores que visam auxiliar os estudantes na construção do seu conhecimento.

As ferramentas utilizadas nos softwares citados podem ser implementados a partir da extensão da classe Ferramenta representada no MC. Essa poderá fazer o papel de artefato mediador entre o sujeito e a comunidade, com o objeto de estudo.

Um outro conceito apresentado diz respeito a colaboração. Este se caracteriza no MC pela classe Comunidade e a Classe DivisãoTrabalho, pois há sempre mais de um membro vinculado a atividade e o desenvolvimento da mesma se dá através

da colaboração de cada um para a transformação do objetivo em resultado.

Um ponto interessante e muito destacado no capítulo 2 é a questão da intencionalidade no aprendizado. Sabe-se que em toda atividade, educacional ou não, ocorre algum tipo de conhecimento. Entretanto, o que caracteriza esse aprendizado no MC é o envolvimento do professor ou responsável no uso do software desenvolvido sob o framework proposto, pois ele se torna o responsável por mediar e promover a intencionalidade do aprendizado.

Além dos softwares educacionais apresentados, os frameworks pesquisados e descritos no capítulo 4 contém características que podem ser aplicadas no MC, como: as ferramentas que também são utilizadas no Média Fusion e Dinosaur Canyon; e a colaboração que caracteriza os framework apresentados, pois, o MC tem o conceito de aprendizagem colaborativa desenvolvida em sua própria estrutura.

5.7 Conclusão

Neste capítulo foram apresentados os elementos da TA com uma visão computacional, os níveis que compõe esta teoria e a adaptação realizada para projetar o framework proposto.

Além disso, foi discutido também as possibilidade de uso deste modelo, bem como, as possibilidades de utilização do conceito de CSCL aplicado ao MC, seguido das implicações educacionais que podem ser utilizadas no modelo desenvolvido.

A partir das definições geradas no MC, foi projetado um framework para ambientes de aprendizagem colaborativa, que é apresentado no capítulo 6.

Capítulo 6

Framework Proposto

6.1 Introdução

Esse capítulo apresenta o FAAC (Um Framework para Ambientes de Aprendizagem Colaborativa) que é a união entre uma técnica computacional denominada Framework Orientado a Objetos descrito no capítulo 3 e o Modelo Computacional (MC) baseado na Teoria da Atividade apresentado no capítulo 5.

O FAAC foi projetado visando facilitar a criação de aplicações de ambientes de aprendizagem colaborativa, que é um aspecto vigente e relevante nos softwares de cunho educacional.

A escolha dessa técnica ocorreu pelo potencial de reutilização oferecido por ela, em o auxílio aos desenvolvedores que podem gerar as aplicações a partir do domínio definido no framework sem iniciar seu trabalho do zero.

6.2 Descrição do FAAC

O primeiro passo para modelar um Framework Orientado a Objetos é definir seu domínio. Como foi apresentado no capítulo 3, existem algumas maneiras de chegar a um domínio de problema. Entretanto, no âmbito deste trabalho os requisitos para classificar este domínio foram obtidos a partir do MC, apresentado no capítulo 5.

Após a definição do domínio foi desenvolvido o diagrama de classe, no qual cada elemento definido no MC tornou-se uma classe do FAAC.

A identificação dos pontos flexíveis do FAAC segue algumas convenções criadas por Silva [Silva 2000], a saber: a Redefinibilidade de classe e a Essencialidade de classe ¹.

- A Redefinibilidade de classe estabelece quais as classes do framework que poderão ser estendidas, ou seja, as classes que podem ou não originar subclasses. A Redefinibilidade será representada com o estereótipo ;R_c nos diagramas.
- A Essencialidade de classe depende diretamente do primeiro item, pois, apenas as classes redefiníveis podem ser consideradas essenciais. Essa notação garante que esta classe será obrigatoriamente redefinível. A Essencialidade será representada pelo estereótipo ;E_c nos diagramas.

6.2.1 Diagrama Arquitetural do FAAC

A Figura 6.1 apresenta o diagrama de classes que contém as principais classes do FAAC com os estereótipos citados acima. Entretanto, para o desenvolvimento do FAAC, duas classes foram adaptadas no diagrama, a classe Atividade e a classe Histórico.

A classe **Atividade** é responsável por centralizar todos os outros elementos da TA. Além desta, uma outra classe que não tem menção direta na estrutura analisada é a classe **Historico**.

A classe **Historico**, embora não seja um elemento do triângulo da atividade, tem um papel fundamental no desenvolvimento de softwares para aprendizagem colaborativa, pois, armazena as informações de cada tarefa realizada individualmente. Assim, o responsável pela atividade, no caso o professor, pode acompanhar como as tarefas estão sendo desenvolvidas, podendo atuar como mediador no processo de aprendizagem.

¹Esses requisitos também foram utilizados por Beutler [Beutler 2003]

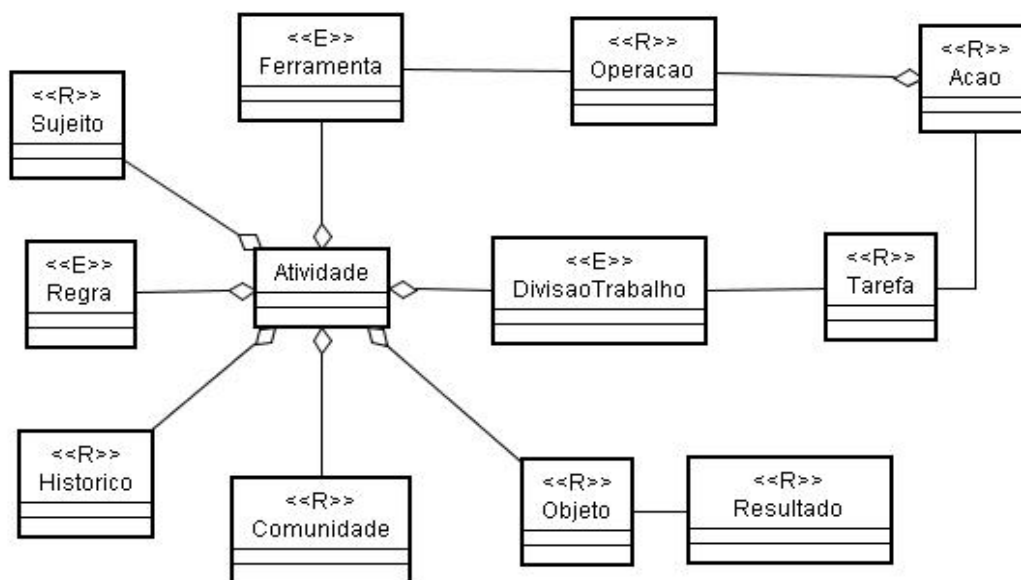


Figura 6.1: Diagrama Arquitetural do FAAC

A classe *Sujeito*, adaptada da estrutura da TA, representa os sujeitos que poderão fazer parte da atividade, sendo que, para entrar na comunidade esses devem respeitar as regras definidas pela comunidade.

A *Regra* é classe responsável pela mediação entre o sujeito e a comunidade, apresentando as permissões e restrições dos mesmos.

A classe *Comunidade* tem a função de criar as regras que serão necessárias para a execução da atividade, além de atribuir os papéis a cada membro da mesma.

A classe *DivisaoTrabalho* faz o papel organizacional dentro da atividade, criando os papéis dos sujeitos e atribuindo a cada papel as tarefas existentes na atividade.

A classe *Tarefa* descreve o primeiro elemento dos níveis de atividade que foram estendidos no triângulo da atividade. Através de cada tarefa pode existir uma ou mais ações, representadas pela classe *Acao*, que é responsável por determinar como

cada tarefa será realizada. E a classe *Operacao* verifica os artefatos disponíveis para realização das ações.

A classe ***Ferramenta*** fornece uma estrutura básica para o desenvolvimento de artefatos para transformação do objeto da aprendizagem no resultado final.

A classe ***Objeto***, além do elemento representado no triângulo da atividade, materializa um princípio apresentado na seção 2.5 do capítulo 2: o princípio de Orientação a Objetos. Esse deixa margem para dois sentidos, a saber: Objeto e Objetivo. A fim de esclarecer essa ambigüidade optou-se por utilizar a definição de [Bellamy 1997], apresentada na seção 2.5 no capítulo 2. A partir dessa, a modelagem da classe *Objeto* provê tanto a opção de estender a classe e atribuir ao objeto seus próprios atributos, como também possibilitar que este seja uma descrição do objetivo da atividade.

6.3 Ambiente de Atividades Colaborativa

No protótipo desenvolvido, o professor é o responsável pelo cadastro da atividade. Após a escolha do objeto de estudo, são definidas quais as ferramentas que irão transformar esse objeto, o papel de cada membro dentro da comunidade, as ações e as operações de cada um, as quais poderão ser consultadas durante todo o processo pelos membros da comunidade.

Este protótipo foi projetado em duas etapas: a primeira é o cadastro da atividade e a segunda etapa é a realização da mesma.

O cadastro das atividades é discutido entre os alunos e o professor, que são os atores deste sistema. O professor entra no sistema e cadastra um nome para a atividade, em seguida, através das reflexões do grupo, ele cadastra um objetivo. Após a escolha do objetivo, o professor cadastra as ferramentas, e em consenso com os envolvidos na atividade, também é realizado o cadastro das regras que deverão ser respeitadas, além dos papéis que podem fazer parte da execução da atividade em questão. A primeira pode ser observada na Figura 6.2.

Figura 6.2: Cadastro da Atividade

Na tela Realizar Atividade, cada aluno irá escolher a atividade que irá desenvolver e o papel que irá desempenhar dentro desta comunidade, alguns exemplos pode ser observado na 6.3.

Figura 6.3: Realizar Atividade

Após a escolha da atividade, o aluno poderá observar o objetivo da comunidade, as regras que foram definidas anteriormente e o papel escolhido por ele. Cada papel está diretamente ligado a uma tarefa. Cada tarefa é composta por uma ação que

materializa o “como” será realizada a mesma, que por sua vez consiste em operações que serão utilizadas para a realização das ações. Entretanto, a operação depende das ferramentas que estiverem disponíveis no sistema. A realização da atividade é representada pela Figura 6.4.

Teoria da Atividade Bem vindo Paula

Atividade Realizar Atividade

Realizar

Histórico

Professor

Atividade

Usuários

Ferramenta

Atividade: Caçada Primitiva
Objetivo: Obter Alimento
Regras: Caçar uma espécie de animal.

Sujeito: Paula

Batedor

Tarefa	Ação	Operação	Ferramenta	Excluir
				X

Resultado:

Salvar

Figura 6.4: Resultado da Atividade1

É importante destacar que a atividade só será iniciada após a definição do objeto de aprendizagem, definido através de reflexões e discussões entre os estudantes e o professor. Assim, somente após esta definição é que os outros elementos da aplicação serão discutidas, como: qual a ferramenta que será utilizada para transformar o objeto; quais as regras para entrar na comunidade dessa atividade; quais os papéis de cada um na atividade e quais as ações e operações de cada membro.

As ferramentas, previamente cadastradas, aparecem no campo Ferramenta Operação para que possa ser selecionada. Após a escolha da ferramenta o aluno contará com um campo que pode ser descrito a ação referente a operação e os comentários sobre sua tarefa, sua ação e operação dentro da atividade.

O resultado é a transformação do próprio objeto, este que pode ser apresentado através das descrições dos alunos e podem ser visualizados na Figura 6.5.

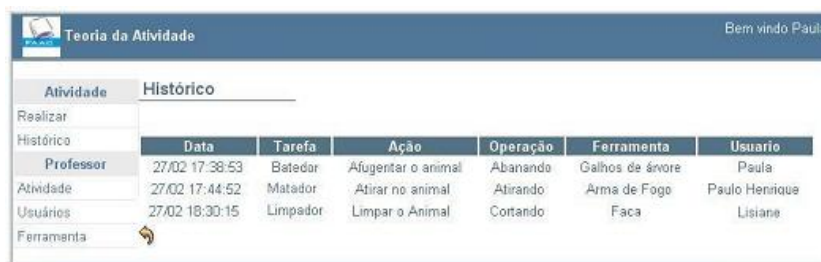
Figura 6.5: Resultado da Atividade2

Os comentários, bem como, o papel que cada um escolheu, a ação, a operação e a ferramenta podem ser acessados por qualquer membro da comunidade através do histórico da atividade. Na Figura 6.6 é apresentado o histórico das atividades realizadas no ambiente.

Figura 6.6: Histórico da Atividade1

O aluno ou professor pode selecionar a atividade e escolher, através do nome do membro da comunidade, qual o histórico que será visualizado. Assim, o professor poderá ter um maior acompanhamento da realização de cada atividade. A Figura

6.7 representa o histórico com as informações definidas por aluno.



Atividade	Histórico					
Realizar						
Histórico						
Professor	27/02 17:38:53	Batedor	Afugentar o animal	Abanando	Galhos de árvore	Paula
Atividade	27/02 17:44:52	Matador	Atirar no animal	Atirando	Arma de Fogo	Paulo Henrique
Usuários	27/02 18:30:15	Limpador	Limpar o Animal	Cortando	Faca	Lisiane
Ferramenta						

Figura 6.7: Histórico da Atividade2

6.4 Conclusão

No presente capítulo foi apresentada uma descrição do framework proposto, bem como, um diagrama de classes representando sua estrutura. Cada classe do FAAC faz menção aos conceitos definidos no MC a partir da análise da Teoria da Atividade.

Ainda neste capítulo foi apresentado um protótipo de um ambiente de aprendizagem desenvolvido sob o FAAC. Neste ambiente o professor tem um papel fundamental na medição entre o aluno e o artefato mediador, pois, ele é o responsável por cadastrar as atividades a partir de discussões e reflexões realizadas entre o grupo.

Capítulo 7

Conclusão

A problemática que deu início a esta pesquisa foi o fato da questão pedagógica estar implícita nos softwares educacionais encontrados no mercado. Isso ocorre pelo fato dos desenvolvedores destes normalmente serem profissionais da área computacional. Além disso, em consequência da formação distinta dos profissionais envolvidos neste processo, há uma dificuldade na comunicação entre desenvolvedores de softwares educacionais e os profissionais da educação.

Normalmente os softwares educacionais encontrados descrevem o modelo educacional comportamentalista, baseado em estímulo e respostas. Entretanto, os softwares educacionais vigentes tendem a enfatizar ambientes de aprendizagem colaborativa.

A Teoria da Atividade é considerada multidisciplinar e esta sendo utilizada na informática na educação para o desenvolvimento de softwares de aprendizagem colaborativa apoiadas por computador.

A fim de amenizar a falta de conhecimento dos desenvolvedores sobre teorias de aprendizagem, foi desenvolvido um Modelo Computacional que traduz a TA para um correlato computacional da mesma. A partir deste modelo, foi projetado um framework para ambientes de aprendizagem colaborativo (FAAC) e desenvolvido um protótipo para implementar as funcionalidades básicas apresentadas nele.

Os softwares desenvolvidos sob o FAAC podem possibilitar aos professores a atuação como mediadores dentro da sala de aula, dando a dinâmica e propondo as atividades que devem ser desenvolvidas, trazendo para o aluno um ambiente propício para usar a sua criatividade.

Entretanto, no decorrer desta pesquisa algumas limitações foram observadas, como por exemplo o fato da Teoria da Atividade não ter uma especificação para educação. Pode-se dizer que a TA ainda é pouco explorada na área educacional o que restringe o número de publicações e aplicações nesta área.

Uma outra limitação diz respeito ao fato da utilização de uma teoria para estabelecer o domínio do FAAC, pois, sua abstração, o que é um ponto positivo em se tratando de framework, acaba dificultando o trabalho do desenvolvedor que terá um maior esforço na utilização do mesmo.

7.1 Trabalhos Futuros

No escopo desta pesquisa alguns pontos podem ser apresentados como trabalhos futuros, entre eles pode-se destacar a implementação de outros frameworks utilizando o Modelo Computacional desenvolvido, bem como implementar outras aplicações para validar as possibilidades de uso do FAAC.

Um outro ponto interessante, diz respeito aos níveis da atividade, atividade, ação e operação. Estes níveis têm uma relação dinâmica entre seus elementos, ou seja, uma atividade pode estar em momento no nível de ação, que é o planejamento de como realizar a atividade ou no nível de operação, onde a ação já esta internalizada.

Entretanto, na presente dissertação, a proposta de utilização dos níveis deu-se de maneira estática, ou seja, não havendo a variação entre os níveis operação, ação e atividade, pois houveram limitações em implementar a dinâmica proposta pela teoria.

Porém, a identificação de algumas atividades que possam concretizar a dinamicidade deste níveis poderão ser definidas futuramente devido limitações destacadas anteriormente.

Referências Bibliográficas

- [Aranha 1996]ARANHA, M. L. A. *Filosofia da Educação*. [S.l.]: Editora Moderna, 1996.
- [Bannon 1997]BANNON, L. *Activity Theory - Interaction Design Centre*. 1997. Acessado em 20/01/2004 URL: <http://www-sv.cict.fr/cotcos/pjs/TheoreticalApproaches/Activity/ActivitypaperBannon.htm>.
- [Bellamy 1997]BELLAMY. *CONTEX AND CONSCIOUSNESS - Designing Educational Technology: Computer Mediated Change. chapter6*. [S.l.]: MIT Press, 1997.
- [Beutler 2003]BEUTLER, D. L. *HYPERTOOLBUILDER: Um framework para a criação de ferramentas de autoria para documentos multimídia*. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, 2003.
- [BOCK 2004]BOCK, A. M. B. A perspectiva sócio-histórica de leontiev e a crítica à naturalização da formação do ser humano: A adolescência em questão. *Cadernos CEDES*, 2004. Acessado em 20/09/2004 URL: <http://www.scielo.br/pdf/ccedes/v24n62/20090.pdf>.
- [Bosch et al. 1997]BOSCH, J. et al. *Object Oriented Frameworks: Problems and Experiences*. 1997.
- [Bourguin 2000]BOURGUIN, G. *Un support informatique à l'activité coopérative fondé sur la Théorie de l'Activité*. Tese (Doutorado) — L'Universite des Sciences et Technologies de Lille, 2000. Acessado em 20/07/2004 URL: <http://lil.univ-littoral.fr/~bourguin/pagesweb/TheseGregoryBourguin.pdf>.

- [Candida 2002]CANDIDA, M. *Matemática com software em lugar da lousa*. 2002. Acessado em 01/12/2004 URL: http://www.unicamp.br/unicamp/unicamp_hoje/ju/agosto2002/unihoje_jul85pag5b.html.
- [Chabert 1997]CHABERT, A. *Habanero and the VRteach Educators*. 1997. Acessado em 28/06/2004 URL: <http://www.isrl.uiuc.edu/isaac/Habanero/Whitepapers/ecscw-habanero.html>.
- [Chabert, Jackson e Petrovicz 1997]CHABERT, A.; JACKSON, E. G. L.; PIETROVICZ, S. *NCSA Habanero-Synchronous collaborative framework and environment*. 1997. Acessado em 20/01/2004 URL: <http://www.isrl.uiuc.edu/isaac/Habanero/>.
- [Davis e Oliveira 1994]DAVIS, C.; OLIVEIRA, Z. M. R. *Psicologia na educação*. 2. ed. [S.l.]: Cortez, 1994. ISBN 85-249-0273-6.
- [Duarte 2004]DUARTE, N. Formação do indivíduo, consciência e alienação: O ser humano na psicologia de a. n. leontiev. *Cadernos CEDES*, v. 24, n. 62, 2004. Acessado em 20/09/2004 URL: <http://www.scielo.br/pdf/ccedes/v24n62/20091.pdf>.
- [Engestrom e Miettinen 1999]ENGESTROM, Y.; MIETTINEN, R. *Perspectives on Activity Theory*. [S.l.]: Cambridge University Press, 1999.
- [Fayad e Schmidt 1997]FAYAD, M.; SCHMIDT, D. C. Object-oriented application frameworks. *Communicação in the ACM*, ACM Press, v. 40, n. 10, p. 32–38, 1997. ISSN 0001-0782. Acessado em 21/08/2003 URL: <http://doi.acm.org/10.1145/262793.262798>.
- [Gall e Hauck 1997]GALL, U.; HAUCK, F. J. Promondia: A Java-based framework for real-time group communication in the Web. *Computer Networks and ISDN Systems*, v. 29, n. 8–13, p. 917–926, 1997. Acessado em 20/01/2004 URL: citeseer.nj.nec.com/gall97promondia.html.

- [Gamma et al. 1995]GAMMA, E. et al. *Design Patterns: Elements of Reusable Object-Oriented Software*. New York, NY: Addison-Wesley, 1995. (Addison-Wesley). ISBN 0-201-63361-2.
- [Habanero 1996]HABANERO. *The Board of Trustees of the University of Illinois*. 1996. Acessado em 28/06/2004 URL: <http://www.isrl.uiuc.edu/isaac/Habanero>.
- [Hautamaki 1997]HAUTAMAKI, J. *A SURVEY OF FRAMEWORKS*. março 1997. Acessado em 09/06/2003 URL: <http://www.cs.uta.fi/reports/pdf/A-1997-3.pdf>.
- [Hsiao 2001]HSIAO, J. W. D. L. *CSCL Theories*. 2001. Cessado em 20/07/2004 URL: <http://www.edb.utexas.edu/csclstudent/Dhsiao/theories.html>.
- [Johnson e Foote 1988]JOHNSON, R.; FOOTE, B. Designing reusable classes. *Journal of Object-Oriented Programming*, Volume 1, n. Number 2, p. pages 22–35, June/July 1988. Acessado em: 10/06/2003 URL: <http://www.laputan.org/drc/drc.html>.
- [Johnson 1997]JOHNSON, R. E. Frameworks = (components + patterns). *ACM*, 1997. Acessado em: 21/06/2004 URL: <http://www-lifia.info.unlp.edu.ar/poo99/Frameworks-patterns.pdf>.
- [Kaptelinin e Nardi 1997]KAPTELININ, V.; NARDI, B. A. *Activity Theory: Basic Concepts and Applications*. 1997. Acessado em 20/01/2004 URL: <http://www.acm.org/sigchi/chi97/proceedings/tutorial/bn.htm>.
- [Komosinski 2000]KOMOSINSKI, L. J. *Um Novo Significado para a Educação Tecnológica Fundamentado na Informática como Artefato Mediador da Aprendizagem*. Tese (Doutorado) — Universidade Federal de Santa Catarina, Dezembro 2000.
- [Kuutti 1997]KUUTTI, K. *CONTEX AND CONSCIOUSNESS - Activity Theory and Human-Computer Interaction. chapter2*. [S.l.]: MIT Press, 1997.

- [Leontiev 1978]LEONTIEV, A. *O Desenvolvimento do Psiquismo*. [S.l.]: Moraes, 1978.
- [Mattsson 1996]MATTSSON, M. *Object-Oriented Frameworks A survey of methodological issues*. Dissertação (Mestrado) — Department of Computer Science and Business Administration University College of Karlskrona/Ronneby, 1996. Acessado em 16/07/2003 URL: <http://citeseer.nj.nec.com/cache/papers/cs/3494/http%3A%2F%2FzSzzSzbilbo.ide.hk-r.se%3A8080zSz~michaelmzSzthesiszSzmatsson-thesis.pdf/mattsson96objectoriented.pdf>.
- [Moll 1996]MOLL, L. *Vygotsky e a educação: Implicações pedagógicas da psicologia sócio-histórica*. [S.l.]: Artes Médicas, 1996.
- [Oliveira 1993]OLIVEIRA, M. K. *Vygotsky - Aprendizado e Desenvolvimento um processo Socio Historico*. [S.l.]: Editora Scipione, 1993. (Pensamento e Acao no Magisterio).
- [Pree 1997]PREE, W. *Component Based Software Development - A New Paradigm in Software Engineering*. 1997. Acessado em 24/07/2003 URL: <http://www.exciton.cs.rice.edu/comp410/frameworks/Pree/J009.pdf>.
- [Pree 2000]PREE, W. *Hot-Spot-Driven Framework Development*. 2000. Acessado em 24/07/2004 URL: <http://www.cs.vu.nl/~ralf/oo/resources/pree2000.pdf>.
- [Pressman 1995]PRESSMAN, R. *Engenharia de Software*. [S.l.]: Makron Books, 1995.
- [Prieto-Díaz e Arango 1991]PRIETO-DÍAZ, R.; ARANGO, G. Domain analysis and software systems modeling. *IEEE Computer Society Press*, 1991.
- [Rego 1999]REGO, T. C. *Vigotsky: Uma perspectiva histórico-cultural da educação*. [S.l.]: Editora Vozes, 1999. (94-4445). ISBN 85.326.1345-4.
- [Roberts e Johnson 1996]ROBERTS, D.; JOHNSON, R. Evolving frameworks: A pattern language for developing object-oriented frameworks. *Proc. of PLoP'96, Third Annual*

- Conference on the Pattern Languages of Programs*, 1996. Acessado em: 20/06/2003
URL: <http://st-www.cs.uiuc.edu/users/droberts/evolve.html>.
- [Rovai 2002]ROVAI, A. *Building Sense of Community at a Distance*. 2002. International Review of Research in Open and Distance Learning.
- [Santos 1999]SANTOS, N. *Desenvolvimento de Software Educacional*. Abril 1999. Acessado em 05/05/2004 URL: http://www.ime.uerj.br/professores/neidenew/Des_Soft.html.
- [Silva 2000]SILVA, R. P. *Suporte ao Desenvolvimento e uso de Frameworks e Componentes*. Tese (Doutorado) — Universidade Federal do Rio Grande do Sul, 2000.
- [Silva 2003]SILVA, R. P. *Engenharia de Software*. 2003.
- [Silva 2004]SILVA, R. P. *Desenvolvimento Orientado a Objetos com Frameworks, Padrões e Componentes*. 2004.
- [Taligent 1994]TALIGENT. *Building object oriented frameworks*. 1994. Acessado em 08/08/2003 URL: <http://www-106.ibm.com/developerworks/java/library/j-oobuilding>.
- [Taligent 1995]TALIGENT. *Leveraging object oriented frameworks*. 1995. Acessado em 08/08/2003 URL: <http://www-106.ibm.com/developerworks/java/library/j-ooleveraging>.
- [Technology e Erlangen 1997]TECHNOLOGY, S.; ERLANGEN, R. E. C. *Promondia - A Java-based framework for group communication*. 1997. Acessado em 20/05/2004 URL: <http://www4.informatik.uni-erlangen.de/Projects/SunTREC/Promondia>.
- [Uchôa 2001]UCHÔA, K. C. A. *Construtivismo em Piaget*. 2001. Acessado em 20/02/2004 URL: <http://www.comp.ufla.br/~kacilene/educacao/piaget.html>.

[Valente 1999]VALENTE, J. A. *O Computador na Sociedade do Conhecimento*. [S.l.]: OEA-NIED/UNICAMP, 1999.

[Valente 2000]VALENTE, J. A. *VISÃO ANALÍTICA DA INFORMÁTICA NA EDUCAÇÃO NO BRASIL: a questão da formação do professor*. 2000. Acessado em 25/11/2003 URL: http://www.inf.ufsc.br/sbc_ie/revista/nr1/valente.htm.

[Vygotsky 1984]VYGOTSKY, L. S. *A formação social da mente*. [S.l.]: M. Fontes, 1984.